université
PARIS-SACLAY

Data Integration: A perpetually evolving challenge for new research perspectives

**Habilitation à diriger des recherches
de l'Université Paris-Saclay**

**prévue d'être présentée et soutenue à Gif-sur-Yvette,
le 16 Septembre 2025, par**

Francesca BUGIOTTI

Composition du jury

Membres du jury avec voix délibérative

Amel Bouzeghoub Professor, Telecom SudParis, Samovar Laboratory	Examinatrice
Zoubida Kedad Professor, Versailles Saint-Quentin-en-Yvelines University, Paris-Saclay University, DAVID Laboratory	Examinatrice
Mohan C. Mohan Distinguished Professor, IBM/Hong Kong Baptist University	Examinateur
Paolo Papotti Associate Professor, Eurecom	Rapporteur & Examinateur
Pascal Poncelet Professor, Montpellier University, LIRRM Laboratory	Rapporteur & Examinateur
Genoveva Vargas-Solar Full-time researcher, CNRS, LIRIS Laboratory	Rapporteur & Examinatrice

Habilitation à diriger des recherches

ÉCOLE DOCTORALE
Sciences et technologies
de l'information et de
la communication (STIC)

Titre: Integration de données : un défi en constante évolution pour de nouvelles perspectives de recherche

Mots clés: Integration de données, NoSQL, Data pour IA, Métamodèles

Résumé:
Les données sont omniprésentes et peuvent être produites et stockées à partir de tout type de contextes au quotidien. Les structures utilisées pour stocker les données sont hétérogènes, en constante évolution et utilisées par les applications selon des modèles variés. Il est démontré que la valeur essentielle de toute application réside dans sa capacité à accéder à des données utiles et fiables. Leur traitement permet d'extraire de la valeur et des informations essentielles. D'autre part, plus les données sont différentes et hétérogènes, plus leur analyse, leur compréhension et les prédictions possibles sont complexes.

Dans ce travail, je résume une partie de mes contributions de recherche au domaine de l'intégration de données hétérogènes. La première partie présente des scénarios classiques d'intégration de données sur des bases de données NoSQL. La deuxième partie aborde le défi du développement de nouvelles techniques d'intégration conçues pour alimenter les algorithmes d'intelligence artificielle. Je présente aussi les applications issues de multiples domaines utilisés comme études de cas lors du développement des idées de recherche : villes intelligentes, stockage du carbone, profilage utilisateur, reconnaissance musicale et intégration de données médicales. Enfin, la dernière partie examine l'application des techniques d'intégration de données à de nouveaux défis et le développement de nouvelles perspectives axées sur les données pour l'IA.

Title: Data integration: a perpetually evolving challenge for new research perspectives

Keywords: Data integration, NoSQL, Data for AI, Metamodeling

Abstract: Data is everywhere and can be produced and stored from everyday scenarios. The structures used for storing data are heterogeneous, continuously evolving, and consumed by applications according to different patterns. It is demonstrated that most of the value of any application is in the ability to access useful and reliable data. From data processing, we can extract value and insights. In addition, the more different and heterogeneous the data are, the more challenging it is to analyze, understand, and make predictions on top of them.

In the HDR manuscript, I summarize part of my research contributions to the field of heterogeneous data integration. The first part of the manuscript presents classical data integration scenarios on NoSQL databases. The second part addresses the challenge of developing new integration techniques conceived for feeding Artificial Intelligence algorithms. The illustration will focus on applications arising from multiple domains used as case studies during the development of the research ideas: smart cities, carbon storage, user profiling, music recognition, and medical data integration. Finally, the last part investigates the application of data integration techniques to new challenges and the development of new prospective oriented in data for AI.

Contents

1 Introduction **7**

1.1 Overview of the contributions 8

1.2 Organization of the manuscript 10

2 Data integration in NoSQL **13**

2.1 Background and context 14

2.1.1 Running example 14

2.1.2 NoSQL database models 15

2.1.3 Key-value stores 16

2.1.4 Document stores 17

2.1.5 Extensible record stores 18

2.1.6 Comparison 19

2.2 Methodology: the NoAM data model 20

2.3 Discussion and results 22

2.3.1 Conceptual modeling and aggregate design 23

2.3.2 Data representation in NoAM and aggregate partitioning 24

2.3.3 Implementation 27

2.3.4 Experiments 30

2.4 Related work 32

2.5 Conclusion 34

3 Graphs and LLM **35**

3.1 Background and context 36

3.1.1 Dataset 37

3.1.2 Questionnaire 38

3.1.3 Virtual Reality Simulations 39

3.1.4 Interview Transcripts 40

3.2 Methodology: an integrated graph Conceptual Schema 40

3.2.1 Modeling the Conceptual Schema 41

3.2.2 Data Extraction 45

3.2.3 Node Disambiguation 46

3.2.4 Data Integration and Resolution 47

3.2.5 Data for AI: The recommendation System 49

3.3 Results and Discussion 50

3.3.1 Evaluation of the Ordinal Classification Algorithm 51

3.3.2 Demonstration of the Recommendation System Use case 52

3.4 Discussion and Results 53

3.4.1 Named Entity Recognition 53

3.4.2 Data Resolution . . . 55
3.4.3 Key Takeaways and Implications . . . 58
3.5 Related Work . . . 58
3.5.1 Data Modeling using Neo4j Graphs . . . 59
3.5.2 Conceptual Modeling and Artificial Intelligence . . . 59
3.5.3 Advances in Data Modeling and Integration . . . 60
3.6 Conclusion . . . 61
3.A Nature of Data Used . . . 62
3.B Sample Prompts for Data Integration . . . 64
3.C Summary of Data Extracted from Expert Interviews Using LLM . . . 66

4 Data integration in smart cities and energy conversion 67
4.1 Background and context . . . 68
4.1.1 Defining smart city . . . 70
4.1.2 Measuring smart cities . . . 73
4.1.3 Smart city architectures and platforms . . . 78
4.2 Data challenges in the context of smart cities . . . 84
4.2.1 Data availability and quality . . . 84
4.2.2 Data heterogeneity and integration . . . 92
4.2.3 Data management . . . 99
4.2.4 Energy Conversion . . . 105
4.2.5 Publications with Simulated vs. Real Data . . . 106
4.3 Related work . . . 108
4.4 Conclusion . . . 111

5 Time series for AI 113
5.1 Introduction . . . 114
5.2 Time Series and AI . . . 115
5.3 PROCLAIM Overview . . . 116
5.4 Preliminary Phase . . . 118
5.4.1 Column-based data formats . . . 118
5.4.2 Data Type Identification . . . 118
5.5 Attribute Profile Representation . . . 119
5.6 Attribute Labeling . . . 122
5.6.1 Clustering . . . 122
5.6.2 Labeling Function . . . 123
5.7 Experiment Results . . . 124
5.8 Dataset and evaluation metrics for AI . . . 126
5.8.1 Dataset . . . 127
5.8.2 Evaluation Metrics . . . 128
5.9 Data cleaning . . . 129
5.9.1 Clustering marker signatures . . . 129
5.10 Classification operation . . . 130

5.10.1 Baseline . . . 130
5.10.2 Designed Hybrid models . . . 131
5.10.3 Data preparation for DL model . . . 132
5.10.4 DL model training . . . 132
5.10.5 Marker Prediction . . . 133
5.11 Experiments and Results . . . 133
5.11.1 Detailed model performance analysis . . . 134
5.11.2 Sliding Window size . . . 136
5.11.3 Model Interpretability . . . 136
5.12 Related Work . . . 138
5.12.1 Algorithms . . . 140
5.13 Conclusions . . . 140

6 Conclusion and Perspectives **143**

A Curriculum Vitae **149**

B Publications **173**

References **178**

Glossary **219**

Acknowledgement

First of all, I would like to thank Paolo Papotti, Pascal Poncelet, and Genoveva Vargas-Solar for having agreed to report this manuscript, as well as Amel Bouzeghoub, Zoubida Kedad, and Mohan C. Mohan for their presence on the jury. A warm thank to Nicole Bidoit, who guided me in the process and supported me in the finalization of this manuscript. I'm extremely grateful to Tatiana, John, Nicolas, and Zoubida for their support in starting the process and to Claire, who solved the final dilemma of the title. Lastly, I'd like to mention Hélène Bonneau, who is not only guaranteeing the technical part of the defence but is also one of my best advisors.

I would not be here without the help and the guidance of Nacera Seghouani and Barbara Dalena, who shared with me the supervision of the works of Molood and Quentin. A warm thank is reserved for Frédéric Boulanger, who gave me 100% of his trust in the "co-supervision" of Shwetha.

The story of this French HDR started thanks to Ioana, who invited me as PhD student in 2011 and who convinced me to continue my research in France. The journey continued in CentraleSupélec, and I would not have been here without the guidance and support of Dominique, Géraldine, Lina, Safouan, and Yolaine, who always gave me their time and advice.

I want to thank my friends wherever they are in the world for their understanding and support during this long journey. Among them, I want to cite Barbara, Catherine, Claude, Dominique, Elena, Francesca, Géraldine, Hélène, Letizia, Luca, Luigi, Marco, Rosanna, and Valérie: near or far, we are always on the same page.

Finally, a big thank is reserved for my family who are an example of a data integration challenge, welcoming different cultures in different places and always demonstrating patience and love.

1 - Introduction

Data can be produced and stored according to different representations and structures. This heterogeneity is an always-evolving playground set for data-integration research. The key challenge is to ensure that any application using these sources can extract value by combining all of them: no matter their nature and size, no matter how different they become, and no matter how the applications will evolve.

Data integration can be seen as an ever-evolving research challenge that adapts in order to support new data sources, new languages, new formats, and new applications. For the past 15 years, I have been interested in various research problems related to data integration, working in strong collaboration with colleagues and PhD students from companies and research laboratories in France and abroad. My research has focused on (i) the problems related to data integration using a meta model approach; (ii) exploring the potential of Large Language Models (LLMs) to streamline data extraction and resolution processes (iii) those relating to the integration and manipulation of large datasets for running applications using Artificial Intelligence in a real industry real-case scenario.

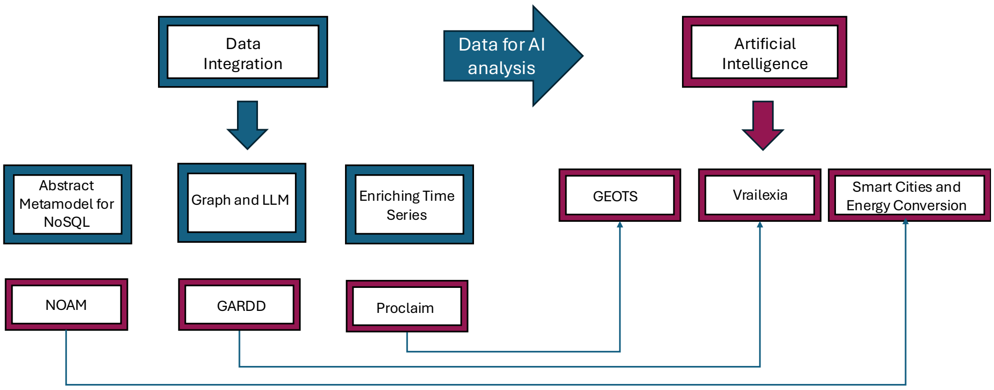


Figure 1.1: Contributions of the manuscript

Data integration enables transparent access to heterogeneous data sources, and an application can access multiple autonomous and heterogeneous (different data models, data schemes, data collection systems) sources uniformly, without affecting the behavior of each source. Having such an integrated and uniform view of data helps any application to run and analyze data also in the dynamic environment of Artificial Intelligence technique. As shown in Figure 1.1 during my research I explored the challenge from both sides. As first I analyzed different techniques for integrating data (contribution in blue) then I explored how such data can be beneficial for multiple applications and context (contributions in red). Such exchange between data and Artificial Intelligence is still an open challenge that is going to guide my future research.

1.1 . Overview of the contributions

The work presented in this manuscript focuses on a selection of my research contributions in data integration. The research deals with the design of data models and their practical usage by different applications. In the rest of this introduction, I present an overview of the research contributions.

Previous Research

During my master's thesis and during my PhD, I explored data integration techniques from a number of perspectives. From the theoretical perspective, we consider Model Management as the framework to formalize translation problems. A schema, an instance of a certain model, will be translated to another schema instance of a target model. We recognize the need for a model-independent solution to schema and data translation and, in general, to model management problems. Hence, we presented MIDST, a tool born from many years of experience on schema and data translation, based on a metalevel approach [J11]. From the performance perspective, we appreciate the value of runtime environments, where translations are not performed out of the system with an import-translate-export process; by contrast, we illustrate, as a novel contribution, MIDST-RT, an evolution of MIDST, where translations are performed at runtime and even generate views of data [J10], [I21]. During these years, the newly introduced NoSQL systems gave me the opportunity to develop SOS, a uniform interface to these systems that also explored indexing strategies [J9], [I19].

Metamodel Data Integration - NOAM for NoSQL

After the achievement of the first promising results with SOS, I continued working on the NoSQL datastores, finalizing the formalization of the research studies on the definition of a general data model whose objective was to express the main characteristics of the NoSQL datastore families. In this research, I continued exploring meta-modeling and the research in collaboration with Luca Cabibbo and Paolo Atzeni (RomaTre University). The results are described in Chapter 2 [J6], [I17]. A second part of this research on NoSQL databases was also developed in collaboration with Ioana Manolescu (INRIA), where we explored how to efficiently distribute data in Amazon DynamoDB [I16], [B1], [I20]. The same research track was extended to MongoDB, where, with an intern, Moditha Hewasinghage, and a PostDoc student, Adnan El Moussawi [I13], we explored how to integrate and distribute data in a cloud environment and in MongoDB [J5], [I12].

Graph data integration and LLM

In this second phase of the studies, I focused on graph data representation and the emerging Large Language Model (LLM) techniques. The dataset used to demonstrate the data modeling and integration methodology represented data of dyslexic students in the context of the Vrailexia project, an EU-funded project led by a consortium of universities across Europe. The three different data sources made available as part of the project were questionnaires, interview transcripts, and virtual reality (VR) simulations. After the definition of an interconnected graph schema, modeled on Neo4j, I explored the potential of Large Language Models (LLMs) to streamline data extraction and resolution processes. The approach aims to address the ongoing challenge of integrating heterogeneous data sources, encouraging advancements in the field of data engineering [J2], [I5], [J3]. This research is illustrated in Chapter 3.

Data Integration for Smart Cities and Energy Conversion

Thanks to the expertise in data integration methodologies all along my research, we decided, in collaboration with Ekaterina Gilman (University of Oulu), to study all the models of data integration useful for smart cities development. This work is described in Chapter 4. The research was extended to energy conversion problems in collaboration with Tatiana Morosuk (TUB - Germany) and an intern, Konstantinos Mira. The research showed how it is possible to use artificial intelligence and machine learning algorithms in energy conversion, management evaluation, and optimization tasks. The work shows how essential it is in this context to give priority to acquiring and integrating real-world experimental and simulated data and adopting standardized, explicit reporting in research publications [J1],[J4].

Time series for AI

In the context of an industry collaboration with the SLB company, I focused on integrating attributes coming from heterogeneous sources. The first part of this research was developed by a Ph.D. student, Molood Arman, and an intern, René Gómez Londoño. In this research context, we developed PROCLAIM (PROfile-based Cluster-Labeling for AttrIbute Matching), a metamodel that performs an automatic, unsupervised clustering-based approach to match attributes of a large number of heterogeneous sources [I8], [I1].

Thanks to the work of a second PhD student, Shwetha Salimath, we integrate time series into this model. We then developed on top of this data a Python library, GeoTS, to apply cutting-edge time series classification models to perform data correlation in a completely automated setting. The development of this library was possible thanks to the flexibility of the newly designed model [I12].

Other work

With respect to the different possible data representations and usages, I also studied data coming from Twitter, defining a new approach that identifies the reputation of an entity on the basis of the set of events it is involved in, by providing a transparent and self-explanatory way for interpreting reputation[J8], [I14], [I15]. Again, focused on the discovery and manipulation of entities, I analyzed data coming from a statistical environment. In collaboration with the Central Bank of Italy, we explored a high-level language, EXL, used for the declarative specification of statistical programs, and a translation into executable form in various target systems is available. The language is based on the theory of schema mappings, in particular those defined by a specific class of TGDs, which we actually use to optimize user programs and facilitate the translation towards several target systems [J7], [I18]. I also explored metal forging processes and the usage of advanced finite element methods. In collaboration with three interns, Shwetha Salimath, Siying Li, and Meduri Venkata Shivadity, we explored the possibility of using a Graph Neural Network-based graph prediction model to act as a surrogate model for parameter search space exploration, and which exhibits a time cost reduced by an order of magnitude[I6], [I7], [I9], [I10]. Finally, with the collaboration of an intern (Dylan Sechet) and a colleague (Matthieu Kowalski), we explored the possibility of developing a reliable coarse-level instrument detection methodology by bridging the gap between detailed instrument identification and group-level recognition, paving the way for further advancements in this domain [I4].

1.2 . Organization of the manuscript

This manuscript is structured along four main chapters (Chapters 2 to 5) devoted to the different lines of research presented above. To keep the manuscript self-contained, each chapter begins with a concise presentation of background concepts. Chapter 6 concludes the manuscript with perspectives about ongoing and future research directions. Finally, Appendix A is a detailed curriculum vitae and Appendix B provides a complete list of publications.

2 - Data integration in NoSQL

NoSQL database systems are today an effective solution to manage large data sets distributed over many servers. In this chapter, we present a high-level data model for NoSQL databases, called NoSQL Abstract Model (NoAM) and show how it can be used as an intermediate data representation in the context of a general design methodology for NoSQL applications having initial steps that are independent of the individual target system. We propose a design process that includes a conceptual phase, as is common in traditional application, followed (and this is unconventional and original) by a system-independent logical design phase, where the intermediate representation is used as the basis for both modeling and performance aspects, with only a final phase that takes into account the specific features of individual systems.

The chapter is adapted from the following papers [a]:

- Paolo Atzeni, Francesca Bugiotti, Luca Cabibbo, Riccardo Torlone: *Data modeling in the NoSQL world*, Comput. Stand. Interfaces 2020
- Paolo Atzeni, Luigi Bellomarini, Francesca Bugiotti, Marco De Leonardis: *Executable schema mappings for statistical data processing*, Distributed Parallel Databases 2018

[a] For all the papers, a link is provided to direct access to the source.

2.1 Background and context 14

2.2 Methodology: the NoAM data model 20

2.3 Discussion and results 22

2.4 Related work 32

2.5 Conclusion 34

The chapter is organized as follows. In Section 2.1, we illustrate the features of the main categories of NoSQL systems arguing that, for each of them, there exists a sort of data model. In Section 2.2 we present NoAM, our system-independent data model for NoSQL databases, and in Section 2.3 we discuss our design methodology for NoSQL databases. In Section 2.4 we briefly review some related literature. Finally, in Section 2.5 we draw some conclusions.

2.1 . Background and context

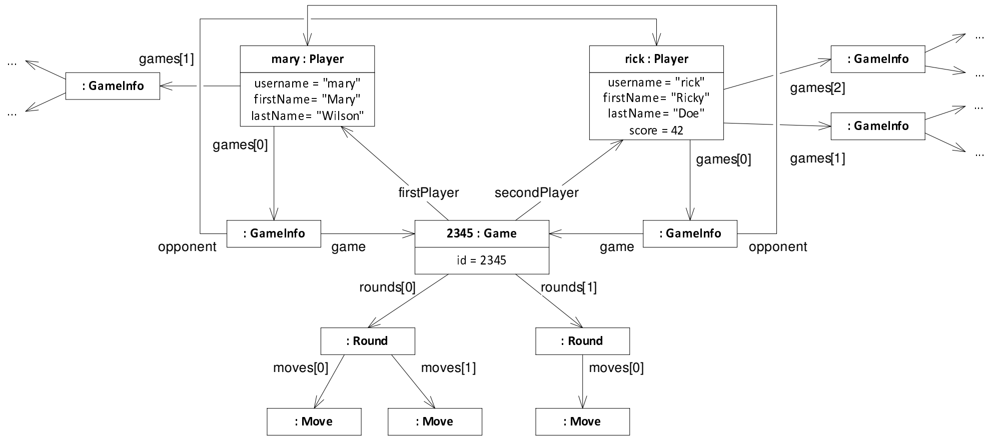


Figure 2.1: Sample application objects.

In this section we briefly present and compare a number of representative NoSQL systems, to make apparent the heterogeneity (as well as the similarities) in the way they organize data and in their programming interfaces. We first introduce a sample application dataset, and then we show how to represent these data in the representative systems we consider.

2.1.1 . Running example

Let us consider, as a running example, an application for an on-line social game. This is indeed a typical scenario in which the use of a NoSQL database is suitable, that is, a simple next-generation Web application (as discussed in the Introduction).

The application should manage various types of objects, including players, games, and rounds. A few representative objects are shown in Figure 2.1. The figure is a UML object diagram. Boxes and arrows denote objects and relationships between them, respectively.

To represent a dataset in a NoSQL database, it is often useful to arrange data in aggregates [138, 183]. Each *aggregate* is a group of related application objects, representing a unit of data access and atomic manipulation. In our example, relevant aggregates are players and games, as shown by closed curves in Figure 2.2. Note that the rounds of a game are grouped within the game itself. In general, aggregates can be considered as complex-value objects [1], as shown in Figure 2.3.

The data access operations needed by our on-line social game are simple read-write operations on individual aggregates; for example, create a new player and retrieve a certain game. Other operations involve just a portion of an aggregate; for example, add a round to an existing game. In general, it is indeed the case that most real applications require only operations that access individual aggregates [87, 92].

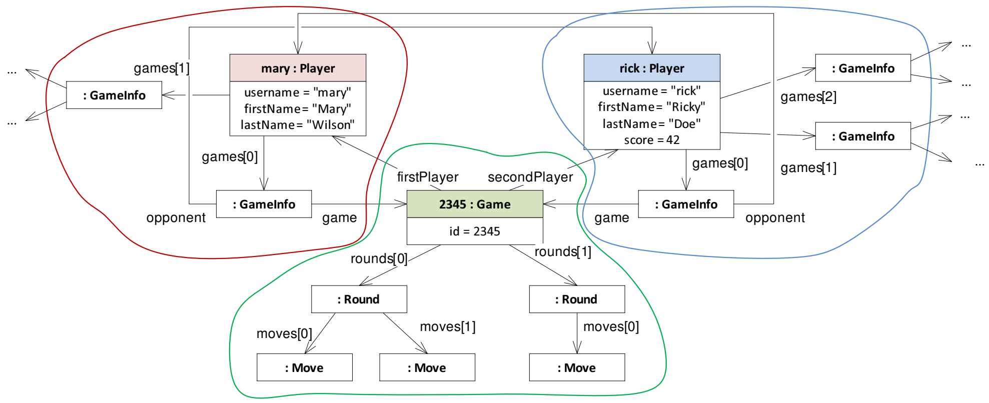


Figure 2.2: Sample aggregates (as groups of objects).

Player:mary : ⟨
username : "*mary*",
firstName : "*Mary*",
lastName : "*Wilson*",
games : {
⟨ *game* : Game:2345, *opponent* : Player:rick ⟩,
⟨ *game* : Game:2611, *opponent* : Player:ann ⟩
}
⟩

Player:rick : ⟨
username : "*rick*",
firstName : "*Ricky*",
lastName : "*Doe*",
score : *42*,
games : {
⟨ *game* : Game:2345, *opponent* : Player:mary ⟩,
⟨ *game* : Game:7425, *opponent* : Player:ann ⟩,
⟨ *game* : Game:1241, *opponent* : Player:johnny ⟩
}
⟩

Game:2345 : ⟨
id : "*2345*",
firstPlayer : Player:mary,
secondPlayer : Player:rick,
rounds : {
⟨ *moves* : . . . , *comments* : . . . ⟩,
⟨ *moves* : . . . , *actions* : . . . , *spell* : . . . ⟩
}
⟩

Figure 2.3: Sample aggregates (as complex values).

2.1.2 . NoSQL database models

NoSQL database systems organize their data according to quite different data models. They usually provide simple read-write data-access operations, which also differ from system to system. Despite this heterogeneity, a few main categories can be identified according to the modeling features of these systems [87, 346]: key-value stores, document stores, extensible record stores, plus others (e.g., graph databases) that are beyond the scope of this chapter.

2.1.3 . Key-value stores

In general, in a *key-value store*, a database is a schemaless collection of key-value pairs, with data access operations on either individual key-value pairs or groups of related pairs.

As a representative key-value store we consider here *Oracle NoSQL* [308]. In this system, *keys* are structured; they are composed of a *major key* and a *minor key*. The major key is a non-empty sequence of strings. The minor key is a sequence of strings. Each element of a key is called a *component* of the key. On the other hand, each *value* is an uninterpreted binary string.

A sample key-value is the pair composed of key */Player/mary/-/username* and value *"mary"*. In the key, symbol '/' separates key components, while symbol '-' separates the major key from the minor key. The distinction between major key and minor is especially relevant to control data distribution and sharding.

In a pair, the value can be either a simple value (such as the string *"mary"*) or a complex value. In the former case, it is common to use some data interchange format (such as XML, JSON, and Protocol Buffers [361]) to represent such complex values.

Oracle NoSQL offers simple atomic access operations, to access and modify individual key-value pairs: $\mathrm{put}(\mathit{key}, \mathit{value})$ to add or modify a key value pair and $\mathrm{get}(\mathit{key})$ to retrieve a value, given the key. Oracle NoSQL also provides an atomic $\mathrm{multiGet}(\mathit{majorKey})$ operation to access a group of related key-value pairs, and specifically the pairs having the same major key. Moreover, it offers an execute operation for executing multiple put operations in an atomic and efficient way (provided that the keys specified in these operations all share a same major key).

The data representation for a dataset in a key-value store can be based on aggregates. These are two common representations for aggregates:

- Representing an aggregate using a single key-value pair. The key (major key) is the aggregate identifier. The value is the complex value of the aggregate. See Figure 2.4a.

- Representing an aggregate using multiple key-value pairs. Specifically, the aggregate is split in parts that need to be accessed or modified separately, and each part is represented by a distinct but related key-value pair. The aggregate identifier is used as major key for all these parts, while the minor key identifies the part within the aggregate. See Figure 2.4b.

The data access operations provided by key-value stores usually enable an efficient and atomic data access to aggregates with respect to both data representations. Indeed, all systems support the access to individual key-value

key (*/major/key/-*)	*value*
/Player/mary/-	*{ username: "mary", firstName: "Mary", ... }*
/Player/rick/-	*{ username: "rick", firstName: "Ricky", ... }*
/Game/2345/-	*{ id: "2345", firstPlayer: "Player:mary", ... }*

a Single key-value pair per aggregate

key (*/major/key/-/minor/key*)	*value*
Player/mary/-/username	*"mary"*
Player/mary/-/firstName	*"Mary"*
Player/mary/-/lastName	*"Wilson"*
Player/mary/-/games[0]	{*game*: "Game:2345", *opponent*: "Player:rick"}
Player/mary/-/games[1]	{*game*: "Game:2611", *opponent*: "Player:ann"}
Player/rick/-/username	*"rick"*
Player/rick/-/firstName	*"Ricky"*
Player/rick/-/lastName	*"Doe"*
Player/rick/-/score	*42*
Player/rick/-/games[0]	{*game*: "Game:2345", *opponent*: "Player:mary"}
Player/rick/-/games[1]	{*game*: "Game:7425", *opponent*: "Player:ann"}
Player/rick/-/games[2]	{*game*: "Game:1241", *opponent*: "Player:johnny"}
Game/2345/-/id	*2345*
Game/2345/-/firstPlayer	"Player:mary"
Game/2345/-/secondPlayer	"Player:rick"
Game/2345/-/rounds[0]	{*moves*: ..., *comments*: ...}
Game/2345/-/rounds[1]	{*moves*: ..., *actions*: ..., *spell*: ...}

b Multiple key-value pairs per aggregate

Figure 2.4: Representing aggregates in Oracle NoSQL.

pairs (useful in the former case) and most of them (such as Oracle NoSQL) provide also the access to groups of related key-value pairs (required in the latter case).

2.1.4 . Document stores

In a *document store*, a database is a set of documents, each having a complex structure and value.

In this category, a widely used system is *MongoDB* [279]. It is an open-source, document-oriented data store that offers a full-index support on any attribute, a rich document-based query API and Map-Reduce support.

In MongoDB, a *database* comprises one or more collections. Each *collection* is a named group of documents. Each *document* is a structured document, that is, a complex value, a set of attribute-value pairs, which can comprise simple values, lists, and even nested documents. Thus, documents are neither freeform text documents nor Office documents. Documents are schemaless, that is, each document can have its own attributes, defined at runtime.

Specifically, MongoDB documents are based on Binary JSON (BSON), a variant of the popular JSON format. Values constituting documents can be of the following types: (i) *basic types*, such strings numbers, dates, and boolean values; (ii) *arrays*, i.e., ordered sequences of values; and (iii) *documents* (or *objects*): a document is a collection of zero or more key-value pairs, where each key is a plain string, while each value is of any of these types. Figure 2.5 shows

```
[
   "username"  : "mary",
   "firstName" : "Mary",
   "lastName"  : "Wilson",
   "games" : {
             [ "id" : "Game:2345", "opponent" : "Player:rick" ],
             [ "id" : "Game:2611", "opponent" : "Player:ann"]
          }
]
```

Figure 2.5: The JSON representation of the complex value of a sample `Player` object.

collection	*document id*	*document*
`Player`	`mary`	`{"_id":"mary", "username":"mary", "firstName":"Mary", ...}`
`Player`	`rick`	`{"_id":"rick", "username":"rick", "firstName":"Rock", ...}`
`Game`	`2345`	`{"_id":"2345", "firstPlayer":"Player:mary", ...}`

Figure 2.6: Representing aggregates in MongoDB.

a JSON representation of the complex value of a sample `Player` aggregate object given in Figures 2.2 and 2.3.

A *main document* is a top-level document with a unique identifier, represented by a special attribute $\_id$, associated to a value of a special type *ObjectId*.

Data access operations are usually over individual documents, which are units of data distribution and atomic data manipulation. The basic operations offered by MongoDB are as follows: insert(*coll*, *doc*) adds a main document *doc* into collection *coll*; and find(*coll*, *selector*) retrieves from collection *coll* all main documents matching document *selector*. The simplest selector is the empty document *{}*, which matches with every document; it allows to retrieve all documents in a collection. Another useful selector is document *{_id:ID}*, which matches with the document having identifier *ID*. There is also an operation to update a document. Moreover, it is also possible to access or update just a specific portion of a document.

In a document store, each aggregate is usually represented by a single main document. The document collection corresponds to the aggregate class (or type). The document identifier *ID* is the aggregate identifier. The content of the document is the complex-value of the aggregate, in JSON/BSON, including also an additional attribute-value pair *{_id:ID}* for the identifier. See Figure 2.6.

Also in this case, the data access operations offered by document stores (such as MongoDB) provide an atomic and efficient data access to aggregates. Specifically, they generally support both operations on individual aggregates, or to specific portions of them, thereof.

2.1.5 . Extensible record stores

In an *extensible record store*, a database is a set of tables, each table is a set of rows, and each row contains a set of attributes (or columns), each with a name *table* **Player**

username	*firstName*	*lastName*	*score*	*games[0]*	*games[1]*	*games[2]*
"mary"	*"Mary"*	*"Wilson"*		{ *game*: ..., *opponent*: ... }	{ ... }	
"rick"	*"Ricky"*	*"Doe"*	42	{ *game*: ..., *opponent*: ... }	{ ... }	{ ... }

table **Game**

id	*firstPlayer*	*secondPlayer*	*rounds[0]*	*rounds[1]*	*rounds[2]*
2345	Player:mary	Player:rick	{ *moves*: ..., *comments*: ... }	{ ... }	

Figure 2.7: Representing aggregates in DynamoDB (abridged).

and a value. Rows in a table are not required to have the same attributes. Data access operations are usually over individual rows, which are units of data distribution and atomic data manipulation.

A representative extensible record store is *Amazon DynamoDB* [18], a NoSQL database service provided on the cloud by Amazon Web Services (AWS). In DynamoDB a database is organized in tables. A *table* is a set of items. Each *item* contains one or more *attributes*, each with a *name* and a *value* (or a set of values). Each table designates an attribute as *primary key*. Items in a same table are not required to have the same set of attributes — apart from the primary key, which is the only mandatory attribute of a table. Thus, DynamoDB databases are mostly schemaless.

Specifically, the primary key is composed of a *partition key* and an optional *sort key*. If the primary key of a table includes a sort key, then DynamoDB stores together all the items having the same partition key, in such a way that they can be accessed in an efficient way.

Distribution is operated at the item level and, for each table, is controlled by the partition key only.

Some operations offered by DynamoDB are as follows: $\mathrm{putItem}(\mathit{table}, \mathit{key}, \mathit{av})$ adds (or modifies) a new item in table *table* with primary key *key*, using the set of attribute-value pairs *av*; and $\mathrm{getItem}(\mathit{table}, \mathit{key})$ retrieves the item of table *table* having primary key *key*. It is also possible to access or update just a subset of the attributes of an item. All these operations can be executed in an efficient way.

In an extensible record store (such as DynamoDB), each aggregate can be represented by a record/row/item. The table corresponds to the aggregate class (or type). The primary key (partition key) is the aggregate identifier. Then, the item can have a distinct attribute-value pair for each top-level attribute of the complex value of the aggregate (or for each major part of the aggregate that needs to be accessed separately). See Figure 2.7.

Again, the data access operations provided by the systems in this category support an efficient data access to aggregates or to specific portions of them.

2.1.6 . Comparison

To summarize, it is possible to say that each NoSQL system provides a number of “modeling elements” to organize data, which can be considered the “data model" of the system. Moreover, the various systems can be effectively classified in a few main categories, where each category is based on "data models" that, even though not identical, do share some similarities. In the next section we show that it is possible to pursue these similarities, thus defining an "abstract data model" for NoSQL databases.

2.2 . Methodology: the NoAM data model

In this section, we present *NoAM* (*NoSQL Abstract Data Model*), a system-independent data model for NoSQL databases. In the following section, we will also discuss how this data model can be used to support the design of NoSQL databases.

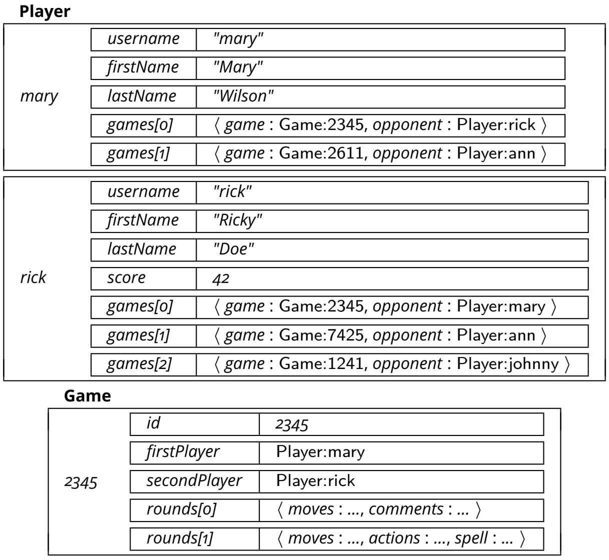
Intuitively, the NoAM data model exploits the commonalities of the data modeling elements available in the various NoSQL systems and introduces abstractions to balance their differences and variations.

A first observation is that all NoSQL systems have a data modeling element that is a data access and distribution unit. By "data access unit" we mean that the system offers operations to access and manipulate an individual unit at a time, in an atomic, efficient, and scalable way. By "distribution unit" we mean that each unit is entirely stored in a server of the cluster, whereas different units are distributed among the various servers. With reference to major NoSQL categories, this element is: (i) a group of related key-value pairs, in key-value stores; (ii) a document, in document stores; or (iii) a record/row/item, in extensible record stores.

In NoAM, a data access and distribution unit is modeled by a *block*. Specifically, a block represents a *maximal* data unit for which atomic, efficient, and scalable access operations are provided. Indeed, while the access to an individual block can be performed in an efficient way in the various systems, the access to multiple blocks can be quite inefficient. In particular, NoSQL systems do not usually provide an efficient "join" operation. Moreover, most NoSQL systems provide atomic operations only over single blocks and do not support the atomic manipulation of a group of blocks. For example, MongoDB [279] provides only atomic operations over individual documents, whereas Bigtable does not support transactions across rows [92].

A second common feature of NoSQL systems is the ability to access and manipulate just a component of a data access unit (i.e., of a block). This component is: (i) an individual key-value pair, in key-value stores; (ii) a field, in document stores; or (iii) a column, in extensible record stores. In NoAM, such a smaller data access unit is called an *entry*.

Finally, most NoSQL databases provide a notion of a collection of data access units. For example, a table in an extensible record store or a document collection in a document store. In NoAM, a collection of data access units is called



Player

mary	*username*	*"mary"*
	firstName	*"Mary"*
	lastName	*"Wilson"*
	games[0]	⟨ *game* : Game:2345, *opponent* : Player:rick ⟩
	games[1]	⟨ *game* : Game:2611, *opponent* : Player:ann ⟩
rick	*username*	*"rick"*
	firstName	*"Ricky"*
	lastName	*"Doe"*
	score	*42*
	games[0]	⟨ *game* : Game:2345, *opponent* : Player:mary ⟩
	games[1]	⟨ *game* : Game:7425, *opponent* : Player:ann ⟩
	games[2]	⟨ *game* : Game:1241, *opponent* : Player:johnny ⟩

Game

2345	*id*	*2345*
	firstPlayer	Player:mary
	secondPlayer	Player:rick
	rounds[0]	⟨ *moves* : ..., *comments* : ... ⟩
	rounds[1]	⟨ *moves* : ..., *actions* : ..., *spell* : ... ⟩

Figure 2.8: A sample database in NoAM.

a *collection*.

According to the above observations, the NoAM data model is defined as follows.

- A NoAM *database* is a set of *collections*. Each collection has a distinct name.
- A collection is a set of *blocks*. Each block in a collection is identified by a *block key*, which is unique within that collection.
- A block is a non-empty set of *entries*. Each entry is a pair $\langle ek, ev \rangle$, where ek is the *entry key* (which is unique within its block) and ev is its value (either complex or scalar), called the *entry value*.

For example, Figure 2.8 shows a possible representation of the aggregates of Figures 2.2 and 2.3 in terms of the NoAM data model. There, outer boxes denote blocks representing aggregates, while inner boxes show entries. Note that entry values can be complex, being another commonality of various NoSQL systems.

Please note that the same data can usually be represented in different ways. Compare, for example, Figure 2.8 with Figure 2.9. We will discuss this possibility in the next section.

In summary, NoAM describes in a uniform way the features of many NoSQL systems, and so can be effectively used, as we show in the next section, for an intermediate representation in a NoSQL database design methodology.

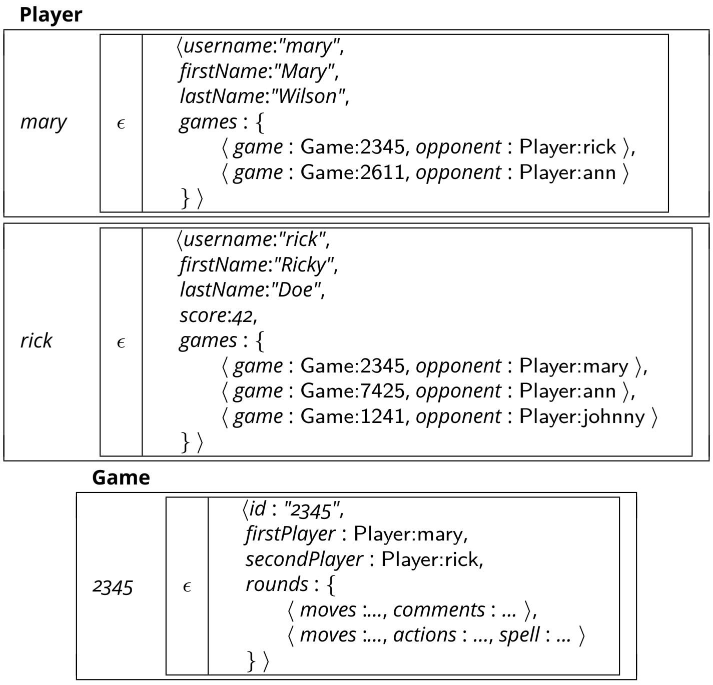

Player
mary
ϵ
⟨username:"mary",
firstName:"Mary",
lastName:"Wilson",
games : {
⟨ game : Game:2345, opponent : Player:rick ⟩,
⟨ game : Game:2611, opponent : Player:ann ⟩
} ⟩
rick
ϵ
⟨username:"rick",
firstName:"Ricky",
lastName:"Doe",
score:42,
games : {
⟨ game : Game:2345, opponent : Player:mary ⟩,
⟨ game : Game:7425, opponent : Player:ann ⟩,
⟨ game : Game:1241, opponent : Player:johnny ⟩
} ⟩
Game
2345
ϵ
⟨id : "2345",
firstPlayer : Player:mary,
secondPlayer : Player:rick,
rounds : {
⟨ moves :..., comments : ... ⟩,
⟨ moves :..., actions : ..., spell : ... ⟩
} ⟩

Figure 2.9: Another NoAM sample database.

2.3 . Discussion and results

The main goal of NoAM is to support a design methodology for NoSQL databases that have initial activities that are independent of the specific target system. In particular, NoAM is used to specify an intermediate, system-independent representation of the application data. The implementation in a target NoSQL system is then a final step, with a translation that takes into account its peculiarities.

The motivations to consider database design for NoSQL systems are as follows. It is important to notice that despite the fact that NoSQL databases are claimed to be "schemaless," the data of interest for applications do show some structure, which should be mapped to the modeling elements (collections, tables, documents, key-value pairs) available in the target system. Moreover, different alternatives in the organization of data in a NoSQL database are usually possible, but they are not equivalent in supporting qualities such as performance, scalability, and consistency (which are typically required when a NoSQL database is adopted). For example, a "wrong" database representation can lead to performance that are worse by an order of magnitude as well as to the inability to guarantee atomicity of important operations.

Specifically, our design methodology has the goal of designing a "good" representation of the application data in a target NoSQL database, and is intended to support major qualities such as performance, scalability, and consistency,

as needed by next-generation Web applications.
The NoAM approach is based on the following main activities:

- *conceptual data modeling* and *aggregate design*, to identify the various entities and relationships thereof needed in an application, and to group related entities into aggregates;

- *aggregate partitioning* and *high-level NoSQL database design*, where aggregates are partitioned into smaller data elements and then mapped to the NoAM intermediate data model;

- *implementation*, to map the intermediate data representation to the specific modeling elements of a target datastore.

In this approach, only the implementation depends on the target datastore. We will discuss the various steps of this approach in the rest of this section.

2.3.1 . Conceptual modeling and aggregate design

The methodology starts, as it is usual in database design, by building a conceptual representation of the data of interest, in terms of entities, relationships, and attributes. (This activity is discussed in most database textbooks, e.g., [58].) Following Domain-Driven Design (DDD [138]), which is a widely followed object-oriented methodology, we assume that the outcome of this activity is a conceptual UML class diagram defining the entities, value objects, and relationships of the application. An *entity* is a persistent object that has independent existence and is distinguished by a unique identifier (e.g., a player or a game, in our running example). A *value object* is a persistent object which is mainly characterized by its value, without its own identifier (e.g., a round or a move). Then, the methodology proceeds by identifying aggregates.
The design of aggregates has the goal of identifying the classes of aggregates for an application, and various approaches are possible. After the preliminary conceptual design phase, entities and value objects are grouped into aggregates. Each *aggregate* has an entity as its root, and it can also contain many value objects. Intuitively, an entity and a group of value objects are used to define an aggregate having a complex structure and value.
The relevant decisions in aggregate design involve the choice of aggregates and of their boundaries. This activity can be driven by the data access patterns of the application operations, as well as by scalability and consistency needs [138]. Specifically, aggregates should be designed as the units on which atomicity must be guaranteed [183] (with eventual consistency for update operations spanning multiple aggregates [324]). In general, it is indeed the case that most real applications require only operations that access individual aggregates [87, 92]. Each aggregate should be large enough so as to include all the data required by a relevant data access operation. (Please note that

NoSQL systems do not provide a “join” operation, and this is a main motivation for clustering each group of related application objects into an aggregate.) Furthermore, to support strong consistency (that is, atomicity) of update operations, each aggregate should include all the data involved by some integrity constraints or other forms of business rules [408]. On the other hand, aggregates should be as small as possible; small aggregates reduce concurrency collisions and support performance and scalability requirements [408].

Thus, aggregate design is mainly driven by data access operations. In our running example, the online game application needs to manage various collections of objects, including players, games, and rounds. Figure 2.2 shows a few representative application objects. (There, boxes and arrows denote objects and links between them, respectively. An object having a colored top compartment is an entity, otherwise it is a value object.) When a player connects to the application, all data on the player should be retrieved, including an overview of the games she is currently playing. Then, the player can select a game to continue, and data on the selected game should be retrieved. When a player completes a round in a game she is playing, then the game should be updated. These operations suggest that the candidate aggregate classes are players and games. Figure 2.2 also shows how application objects can be grouped in aggregates. (There, a closed curve denotes the boundary of an aggregate.)

As we mentioned above, aggregate design is also driven by consistency needs. Assume that the application should enforce a rule specifying that a round can be added to a game only if some condition involving the other rounds of the game is satisfied. An individual round cannot check, alone, the above condition; therefore, it cannot be an aggregate by itself. On the other hand, the above business rule can be supported by a game (comprising, as an aggregate, its rounds).

In conclusion, the aggregate classes for our sample application are **Player** and **Game**, as shown in Figures 2.2 and 2.3.

2.3.2 . Data representation in NoAM and aggregate partitioning

In our approach, we use the NoAM data model (Section 2.2) as an intermediate model between application aggregates (Section 2.3.1) and NoSQL databases (Section 2.1). We represent each class of aggregates by means of a distinct collection, and each individual aggregate by means of a block. We use the class name to name the collection, and the identifier of the aggregate as block key. The complex value of each aggregate is represented by a set of entries in the corresponding block. For example, the aggregates of Figures 2.2 and 2.3 can be represented by the NoAM database shown in Figure 2.8. The representation of aggregates as blocks is motivated by the fact that both concepts represent a unit of data access and distribution, but at different abstraction

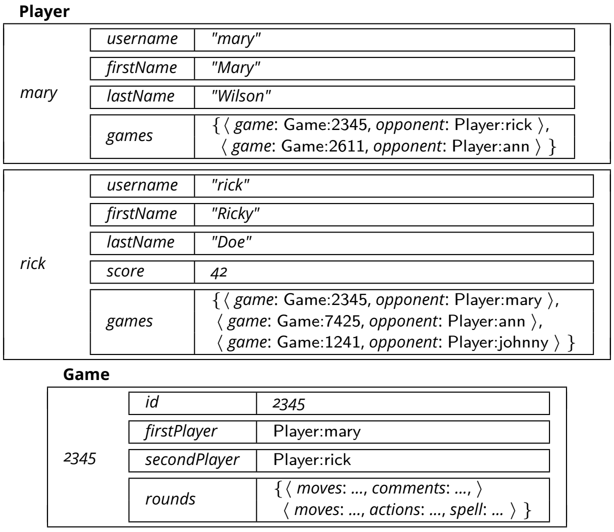

Player
mary
username | "mary"
firstName | "Mary"
lastName | "Wilson"
games | { ⟨ game: Game:2345, opponent: Player:rick ⟩, ⟨ game: Game:2611, opponent: Player:ann ⟩ }
rick
username | "rick"
firstName | "Ricky"
lastName | "Doe"
score | 42
games | { ⟨ game: Game:2345, opponent: Player:mary ⟩, ⟨ game: Game:7425, opponent: Player:ann ⟩, ⟨ game: Game:1241, opponent: Player:johnny ⟩ }
Game
2345
id | 2345
firstPlayer | Player:mary
secondPlayer | Player:rick
rounds | { ⟨ moves: ..., comments: ..., ⟩ ⟨ moves: ..., actions: ..., spell: ... ⟩ }

Figure 2.10: The ETF data representation.

levels. Indeed, NoSQL systems provide efficient, scalable, and consistent (i.e., atomic) operations on blocks and, in turn, this choice propagates such qualities to operations on aggregates.

In general, an application dataset of aggregates can be represented in NoAM database in several different ways. Each *data representation* for a dataset δ is a NoAM database D_δ representing δ. Specifically, the various data representations for a dataset differ only in the choice of the entries used to represent the complex value of each aggregate. We first discuss basic data representation strategies, which we illustrate with respect to the example described in Figure 2.3. We then introduce additional and more flexible data representations.

A simple data representation strategy, called *Entry per Aggregate Object* (*EAO*), represents each individual aggregate using a single entry. The entry key is empty. The entry value is the whole complex value of the aggregate. The data representation of the aggregates of Figure 2.3 according to the EAO strategy is shown in Figure 2.9.

Another data representation strategy, called *Entry per Top-level Field* (*ETF*), represents each aggregate by means of multiple entries, using a distinct entry for each top-level field of the complex value of the aggregate. For each top-level field f of an aggregate o, it employs an entry having as value the value of the field f in the complex value of o (with values that can be complex themselves), and as key the field name f. Figure 2.10 shows the data representation of the aggregates of Figure 2.3 according to the ETF strategy.

As a comparison, we can observe that the EAO data representation uses a block with a single entry to represent the **Player** object having username *mary*, while the ETF representation needs a block with four entries, corresponding to fields *username*, *firstName*, *lastName*, and *games*. Moreover, blocks in EAO do not depend on the structure of aggregates, while blocks in ETF depend on the top-level structure of aggregates (which can be "almost fixed" within each class).

The general data representation strategies we just described can be suited in some cases, but they are often too rigid and limiting. For example, none of the above strategies leads to the data representation shown in Figure 2.8. The main limitation of such general data representations is that they refer only to the structure of aggregates, and do not take into account the data access patterns of the application operations. Therefore, these strategies are not usually able to support the performance of these operations. This motivates the introduction of aggregate partitioning.

We first need to introduce a preliminary notion of *access path*, to specify a "location" in the structure of a complex value. Intuitively, if v is a complex value and w is a value (possibly complex as well) occurring in v, then the access path ap for w in v represents the sequence of "steps" that should be taken to reach the component value w in v. More precisely, an access path ap is a (possibly empty) sequence of *access steps*, $ap = p_1 \, p_2 \ldots p_n$, where each step p_i identifies a component value in a structured value. Furthermore, if v is a complex value and ap is an access path, then $ap(v)$ denotes the component value identified by ap in v.

For example, consider the complex value v_{mary} of the **Player** aggregate having username *mary* shown in Figure 2.3. Examples of access paths for this complex value are *firstName* and *games[0].opponent*. If we apply these access paths to v_{mary}, we access values *Mary* and Player:rick, respectively.

A complex value v can be represented using a set of entries, whose keys are access paths for v. Each entry is intended to represent a distinct portion of the complex value v, characterized by a location in its structure (the access path, used as entry key) and a value (the entry value). Specifically, in NoAM we represent each aggregate by means of a *partition* of its complex value v, that is, a set E of entries that fully cover v, without redundancy. Consider again the complex value v_{mary} shown in Figure 2.3; a possible entry for v_{mary} is the pair $\langle$*games[0].opponent*, Player:rick$\rangle$. We have already applied the above intuition earlier in this section. For example, the ETF data representation (shown in Figure 2.10) uses field names as entry keys (which are indeed a case of access paths) and field values as entry values.

Aggregate partitioning can be based on the following guidelines (which are a variant of guidelines proposed in [58] in the context of logical database design):

- If an aggregate is small in size, or all or most of its data are accessed or modified together, then it should be represented by a single entry.
- Conversely, an aggregate should be partitioned in multiple entries if it is

Game

2345	ϵ	⟨ *id*:2345, *firstPlayer*:Player:mary, *secondPlayer*:Player:rick ⟩
	rounds[0]	⟨ *moves* : …, *comments* : … ⟩
	rounds[1]	⟨ *moves* : …, *actions* : …, *spell* : … ⟩

Figure 2.11: An alternative data representation for games (Rounds).

large in size and there are operations that frequently access or modify only specific portions of the aggregate.

- Two or more data elements should belong to the same entry if they are frequently accessed or modified together.
- Two or more data elements should belong to distinct entries if they are usually accessed or modified separately.

The application of the above guidelines suggests a partitioning of aggregates, which we will use to guide the representation in the target database.

For example, in our sample application, consider the operations involving games and rounds. When a player selects to continue a game, data on the selected game should be retrieved. When a player completes a round in a game she is playing, then the aggregate for the game should be updated. To support performance, it is desirable that this update is implemented in the database just as an addition of a round to a game, rather than a complete rewrite of the whole game. Thus, data for each individual round is always read or written together. Moreover, data for the various rounds of a game are read together, but each round is written separately. Therefore, each round is a candidate to be represented by an autonomous entry. These observations lead to a data representation for games shown in Figure 2.8. However, apart from rounds, the remaining data for each game comprises just a few fields, which can be therefore represented together in a single entry. This further observation leads to an alternative data representation for games, shown in Figure 2.11.

2.3.3 . Implementation

We now discuss how a NoAM data representation can be implemented in a target NoSQL database. Given that NoAM generalizes the features of the various NoSQL systems, while keeping their major aspects, it is rather straightforward to perform this activity. We have implementations for various NoSQL systems, including Cassandra, Couchbase, Amazon DynamoDB, HBase, MongoDB, Oracle NoSQL, and Redis. For the sake of space, we discuss the implementation only with respect to a single representative system for each main NoSQL category. Moreover, with reference to the same aggregate objects of Figures 2.2 and 2.3 we will sometimes show only the data for one aggregate.

Similar representations can be obtained for the other aggregates of the running example.

Key-value store: Oracle NoSQL

In the key-value store Oracle NoSQL [308] (Section 2.1.3), a data representation D for an application dataset can be implemented as follows. We use a key-value pair for each entry $\langle ek, ev \rangle$ in D. The major key is composed of the collection name C and the block key id, while the minor key is a proper coding of the entry key ek (recall that ek is an access path, which we represent using a distinct key component for each of its steps). An example of key is */Player/mary/-/firstName*, where symbol */* separates components, and symbol - separates the major key from the minor key. The value associated with this key is a representation of the entry value ev; for example, *Mary*. The value can be either simple or a serialization of a complex value, e.g., in JSON.

The retrieval of a block can be implemented, in an efficient and atomic way, using a single multiGet operation — this is possible because all the entries of a block share the same major key. The storage of a block can be implemented using various put operations. These multiple put operations can be executed in an atomic way — since, again, all the entries of a block share the same major key.

For example, Figure 2.4b shows the implementation in Oracle NoSQL of the data representation of Figure 2.8. Moreover, Figure 2.4a shows the implementation in Oracle NoSQL of the EAO data representation of Figure 2.9.

An implementation can be considered *effective* if aggregates are indeed turned into units of data access and distribution. The effectiveness of our implementation is based on the use we make of Oracle NoSQL keys, where the major key controls distribution (sharding is based on it) and consistency (an operation involving multiple key-value pairs can be executed atomically only if the various pairs are over a same major key).

More precisely, a technical precaution is needed to guarantee atomic consistency when the selected data representation uses more than one entry per block. Consider two separate operations that need to update just a subset of the entries of the block for an aggregate object. Since aggregates should be units of atomicity and consistency, if these operations are requested concurrently on the same aggregate object, then the application would require that the NoSQL system identifies a concurrency collision, commits only one of the operations, and aborts the other. However, if the operations update two *disjoint* subsets of entries, then Oracle NoSQL is unable to identify the collision, since it has no notion of block. We support this requirement, thus providing atomicity and consistency over aggregates, by always including in each update operation the access to the entry that includes the identifier of the aggregate (or some other distinguished entry of the block).

Extensible record store: DynamoDB

In the extensible record store Amazon DynamoDB ([18], Section 2.1.5), the implementation of a NoAM database can be based on a distinct table for each collection, and a single item for each block. The item contains a number of attributes, which can be defined from the entries of the block for the item.

A NoAM data representation D can be represented in DynamoDB as follows. Consider a block b in a collection C having block key id. According to D, one or multiple entries are used within each block. We use all the entries of a block b to create a new item in a table for b. Specifically, we proceed as follows: (i) the collection name C is used as a DynamoBD table name; (ii) the block key id is used as a DynamoBD primary key in that table; (iii) the set of entries (key-value pairs) of a block b is used as the set of attribute name-value pairs in the item for b (a serialization of the values is used, if needed). For example, Figure 2.7 shows the implementation of the NoAM database of Figure 2.8.

The retrieval of a block, given its collection C and block key id, can be implemented by performing a single getItem operation, which retrieves the item that contains all the entries of the block. The storage of a block can be implemented using a putItem operation, to save all the entries of the block, in an atomic way. It is worth noting that, using operation getItem, it is also possible to retrieve a subset of the entries of a block. Similarly, using operation updateItem, it is also possible to update just a subset of the entries of a block, in an atomic way.

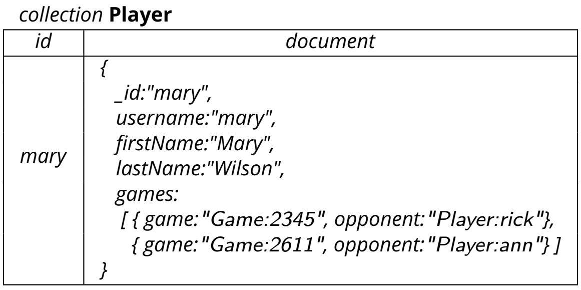
This implementation is also effective, since DynamoDB controls distribution and atomicity with reference to items.

Document store: MongoDB

In *MongoDB* ([100], Section 2.1.4), which is a document store, a natural implementation for a NoAM database can be based on a distinct MongoDB collection for each collection of blocks, and a single main document for each block. The document for a block b can be defined as a suitable JSON/BSON serialization of the complex value of the entries in b, plus a special field to store the block key id of b, as required by MongoDB, *{_id:id}*.

With reference to a NoAM data representation D, consider a block b in a collection C having block key id. If b contains just an entry e, then the document for b is just a serialization of e. Otherwise, if b contains multiple entries, we use all the entries in block b to create a new document. Specifically, we proceed by building a document d for b as follows: (i) the collection name C is used as the MongoDB collection name; (ii) the block key id is used for the special top-level id field *{_id:id}* of d; (iii) then, each entry in the block b is used to fill a (possibly nested) field of document d. See Figure 2.12.

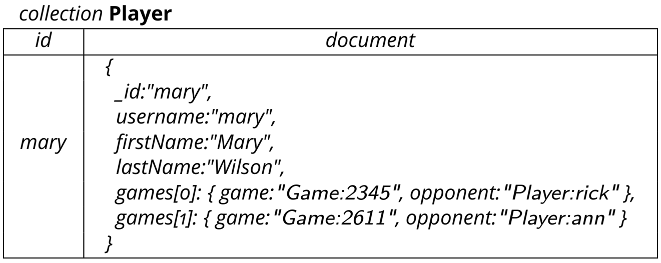
The retrieval of a block, given its collection C and key id, can be implemented



collection **Player**

id	*document*
mary	*{* *_id:"mary",* *username:"mary",* *firstName:"Mary",* *lastName:"Wilson",* *games:* *[{ game: "Game:2345", opponent: "Player:rick"},* *{ game: "Game:2611", opponent: "Player:ann"}]* *}*

Figure 2.12: Implementation in MongoDB.



collection **Player**

id	*document*
mary	*{* *_id:"mary",* *username:"mary",* *firstName:"Mary",* *lastName:"Wilson",* *games[0]: { game: "Game:2345", opponent: "Player:rick" },* *games[1]: { game: "Game:2611", opponent: "Player:ann" }* *}*

Figure 2.13: Alternative implementation in MongoDB.

by performing a find operation, to retrieve the main document that represents all the block (with its entries). The storage of a block can be implemented using an insert operation, which saves the whole block (with its entries), in an atomic way. It is worth noting that, using other MongoDB operations, it is also possible to access and update just a subset of the entries of a block, in an atomic way.

An alternative implementation for MongoDB is as follows. Each block b is represented, again, as a main document for b, but using a distinct top-level field-value pair for each entry in the NoAM data representation. In particular, for each entry (ek, ev), the document for b contains a top-level field whose name is a coding for the entry key (access path) ek, and whose value is either an atomic value or an embedded document that serializes the entry value ev. For example, according to this implementation, the data representation of Figure 2.8 leads to the result shown in Figure 2.13.

2.3.4 . Experiments

We will now discuss a case study of NoSQL database design, with reference to our running example. For the sake of simplicity, we just focus on the representation and management of aggregates for games.

Data for each game include a few scalar fields and a collection of rounds. The important operations over games are: (1) the retrieval of a game, which should read all the data concerning the game; and (2) the addition of a round to a game.

Assume that, to manage games, we have chosen a key-value store as the target system. The candidate data representations are: (i) using a single entry for each game (as shown in Figure 2.9, in the following called EAO); (ii) splitting the data for each game in a group of entries, one for each round, and including all the remaining scalar fields in a separate entry (as shown in Figure 2.11, called Rounds).

We expect that the first operation (retrieval of a game) performs better in EAO, since it needs to read just a key-value pair, while the second one (addition of a round to a game) is favored by Rounds, which does not require to rewrite the whole game.

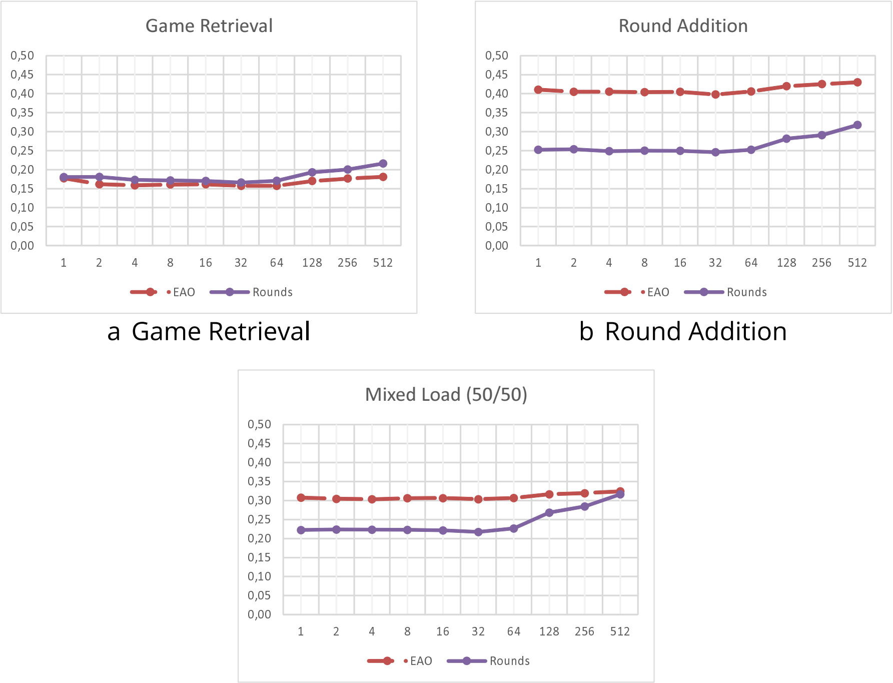
We ran a number of experiments to compare the above data representations in situations of different application workloads. Each game has, on average, a dozen rounds, for a total of about 8KB per game. At each run, we simulated the following workloads: (a) game retrievals only (in random order); (b) round additions only (to random games); and (c) a mixed workload, with game retrieval and round addition operations, with a read/write ratio of 50/50. We ran the experiments using different database sizes, and measured the running time required by the workloads. The target system was Oracle NoSQL, deployed over Amazon AWS on a cluster of four EC2 servers.[1]

The results are shown in Figure 2.14. Database sizes are in gigabytes, timings are in milliseconds, and points denote the average running time of a single operation. The experiments confirm the intuition that the retrieval of games (Figure 2.14a) is always favored by the EAO data representation, for any database size. On the other hand, the addition of a round to an existing game (Figure 2.14b) is favored by the Rounds data representation. Finally, the experiments over the mixed workload (Figure 2.14c) show a general advantage of Rounds over EAO, which however decreases as the database size increases. Overall, it turns out that the Rounds data representation is preferable.

We also performed other experiments on a data representation that does not conform to the design guidelines proposed in this chapter. Specifically, a data representation that divides the rounds of a game into independent key-value pairs, rather than keeping them together in a same block, as suggested by our approach. In this case, the performance of the various operations worsens by at least an order of magnitude. Moreover, with this data representation it is not possible to update a game in an atomic way.

Overall, these experiments show that: (i) the design of NoSQL databases should be done with care as it affects considerably the performance and consistency of data access operations, and (ii) our methodology provides an effective tool for choosing among different alternatives.

[1] This activity was supported by AWS in Education Grant award.



a Game Retrieval

b Round Addition

c Mixed Load

Figure 2.14: Experimental results.

2.4 . Related work

Although several authors have observed that there is a need for data-model approaches to the design and management of NoSQL databases [48, 50, 278], very few works have addressed this issue, especially from a general and system-independent point of view. Indeed, most of them propose a solution to a specific problem in a limited scenario.

For instance, Pasqualin et al. [313] have recently shown how a document-oriented model can be efficiently implemented in a NoSQL document store. Similarly, Olivera et al. [306] and de Lima and Mello [117] have proposed a data-model based methodology for the design of NoSQL document database, whereas Chevalier et al. [98] have addressed the specific problem of leveraging on a document-oriented model for implementing a multidimensional database in a NoSQL document store and in a column-oriented NoSQL database.

Most of the other contributions to data modeling for NoSQL systems come from on-line papers, usually published in blogs of practitioners, that discuss best practices and guidelines for modeling NoSQL databases, most of which are suited only for specific systems. For instance, [224] lists some techniques for implementing and managing data stored in different types of NoSQL systems, while [304] discusses design issues for the specific case of key-value datastores. Similarly, Mior et al. [277] have proposed an approach to the problem of schema design for the specific class of extensible record stores. On the system-oriented side, [94, 229, 174] illustrate design principles for the specific cases of HBase, MongoDB, and Cassandra, respectively. However, none of them tackles the problem from a general perspective, as we advocate in this chapter.

Recently, Ruiz et al. [344] have proposed a reverse engineering strategy aimed at inferring the implicit schema of NoSQL databases. This approach supports the idea that, even in this context, a model-based description of the organization of data is very useful during the entire life-cycle of a data set.

To the best of our knowledge, this chapter presents the first general design methodology for NoSQL systems with initial activities that are independent of the specific target system. Our approach to data modeling is based on data aggregates, a notion that is central in NoSQL databases where application data are grouped in atomic units that are accessed and manipulated together [346]. The notion of aggregate also occurs in other contexts with a similar meaning. For example, in Domain Driven Design [138], a widely followed object-oriented software development approach, an aggregate is a group of related application objects, used to govern transactions and distribution. Also Helland [183] advocates the use of aggregates (there called entities) as units of distribution and consistency. In this framework, Baker et al. [52] propose the notion of entity groups, a set of entities that can be manipulated in an atomic way. They also describe a specific mapping of entity groups to Bigtable [92], which however makes the approach targeted only to a specific NoSQL system. Our approach is based on a more abstract database model, NoAM, and is system independent, as it is targeted to a wide class of NoSQL systems.

The issue of identifying data access units in database design shows some similarities with problems studied in the past, such as: (i) the early works on vertical partitioning and clustering [390], with the idea to put together the attributes that are accessed together and to separate those that are visited independently, and (ii) the more recent approaches to relational (or object-relational) storage of XML documents [143], where various alternatives obviously exist, with tables that can be very small and handle individual edges, or very wide and handle entire paths, and many alternatives in between.

A major observation from [48] is that the availability of a high-level representation of the data remains a fundamental tool for developers and users, since it makes understanding, managing, accessing, and integrating information sources much easier, independently of the technologies used. We have addressed this issue by proposing NoAM, an abstract data model that makes it possible to devise an initial phase of the design process that is independent of any specific system but suitable for each.

Along this line, SOS [47] is a tool that provides a common programming interface towards different NoSQL systems, to access them in a unified way. The interface is based on a simple, high-level common data model which is inspired by those of non-relational systems and provides simple operations for inserting, deleting, and retrieving database objects. However, the definition of tools for data access is complementary to data models and design issues. Finally, Jain et al. discusses the potential mismatch between the requirements of scientific data analysis and the models and languages of relational database systems [212], whereas Alagiannis et al. [8] advocate a new database design philosophy for emerging applications. This chapter tries to provide a contribution to these problems.

2.5 . Conclusion

In this chapter we have discussed how data modeling can be useful in the NoSQL arena. Specifically, we have proposed a comprehensive methodology for the design of NoSQL databases. The methodology relies on an aggregate-oriented view of application data, an intermediate system-independent data model for NoSQL datastores, and finally an implementation activity that takes into account the features of specific systems.

3 - Graphs and LLM

Manually integrating data of diverse formats and languages is vital to many artificial intelligence applications. However, the task itself remains challenging and overly time-consuming.

This chapter highlights the potential of Large Language Models (LLMs) to streamline data extraction and resolution processes. Our approach aims to address the ongoing challenge of integrating heterogeneous data sources, encouraging advancements in the field of data engineering. Applied on the specific use case of learning disorders in higher education, our research demonstrates LLMs' capability to effectively extract data from unstructured sources. It is then further highlighted that LLMs can enhance data integration by providing the ability to resolve entities originating from multiple data sources. The chapter describes how we defined a GrAph Schema foR Dyslexic Disorders (GARDD). and underscores the necessity of preliminary data modeling decisions to ensure the success of such technological applications. By merging human expertise with LLM-driven automation, this study advocates for the further exploration of semi-autonomous data engineering pipelines.

The chapter is adapted from the following papers:

- Adel Remadi, Karim El Hage, Yasmina Hobeika, Francesca Bugiotti: *To prompt or not to prompt: Navigating the use of Large Language Models for integrating and modeling heterogeneous data*, DKE 2024
- Antoine Harfouche, Bernard Quinio, Francesca Bugiotti: *Human-Centric AI to Mitigate AI Biases: The Advent of Augmented Intelligence*, J. Glob. Inf. Manag. 2023
- Karim El Hage, Adel Remadi, Yasmina Hobeika, Ruining Ma, Victor Hong, Francesca Bugiotti: *A multi-source graph database to showcase a recommender system for dyslexic students*, IEEE Big Data 2023

3.1 Background and context 36

3.2 Methodology: an integrated graph Conceptual Schema 40

3.3 Results and Discussion 50

3.4 Discussion and Results 53

3.5 Related Work 58

3.6 Conclusion 61

3.A Nature of Data Used 62

3.B Sample Prompts for Data Integration 64
3.C Summary of Data Extracted from Expert Interviews Using LLM 66

The remainder of the chapter is structured as follows. Section 3.5 introduces the related work that supports the different approaches and strategies considered in our scientific methodology. Section 3.1.1 introduces the different types of structured and unstructured sources that are used in our study. Section 3.2 presents GARDD and details the data modeling choices that served as a foundation for the integration of the heterogeneous sources and the manner in which an LLM can be used to automate the data integration process. Section 3.4 assesses the quality of the proposed automated data integration process and describes some key takeaways and implications. Finally, Section 3.6 summarizes our findings and proposes possible avenues for future research.

3.1 . Background and context

Data Integration is a critical step of any pipeline when considering multiple heterogeneous data sources [125]. In this chapter, we will see how to build GARDD an interconnected graph schema, modeled on Neo4j, a database management system implementing using data sources of different structures and languages and language model applications [249]. Despite considerable advancements in data integration automation, both through traditional semantic techniques [56, 389] and recent language model applications [249], there remains a critical dependency on extensive fine-tuning over large training datasets. The necessity for extensive training stems from the requirement for models to possess a deep comprehension of linguistic subtleties and domain-specific knowledge relevant to the studied use-case [249].
Large Language Models (LLMs) have significantly enhanced the ease with which we can retrieve and interpret data, showcasing the ability to handle a diverse range of tasks. LLMs often require merely one or a few examples to perform tasks, and in certain cases, have outperformed traditional supervised models in terms of effectiveness and efficiency [248, 419]. Despite the potential benefits, [169] points out that the effectiveness of LLMs in data integration, especially in completing complex tasks like entity matching or resolution, remains uncertain. On the other hand, [140] argues that the unique ability of LLMs to understand semantic ambiguities and integrate data from real-world scenarios necessitates a fundamental rethinking of established data management approaches. This perspective underscores the necessity to recognize the potential benefits of incorporating these advanced tools into data management strategies. Considering this, our work investigates the use of LLMs to aid in the

automation of data extraction and integration tasks. The work further investigates the collaborative role that human data modeling design could play to enhance such an automated pipeline. This is done by designing a conceptual schema for a unique and heterogeneous dataset from scratch, elaborating on the importance of the design considerations. Consequently, we were able to use the schema to both guide the prompts fed into the LLM and ensure that the output of the LLM respects the proposed schema. As a result, this work demonstrated that the use of LLMs, guided by prompts that consider human data modeling considerations, is a very encouraging approach to automate the integration of data originating from heterogeneous sources. Hence, the contributions of this work are as follows:

1. Introducing a conceptual schema methodology designed to accommodate a selected dataset composed of multiple sources, each varying in format and language.

2. Automating the extraction of entities from unstructured data sources using a Large Language Model in the context of the defined conceptual schema.

3. Automating the data integration of entities originating from multiple data sources (structured/unstructured data) using a Large Language Model in the context of the GARDD defined conceptual schema.

The introduction of data modeling and integration using LLMs into this work's methodology not only addresses the manual and time-consuming aspects of traditional data integration processes but also addresses the advanced capabilities of LLMs to understand and process language nuances. This approach enables a more efficient and effective integration of diverse data sources, widening the range of possibilities for data integration practices in various fields.

3.1.1 . Dataset

The dataset used to demonstrate the data modeling and integration methodology comes from the Vrailexia project, an EU-funded project comprised of a consortium of universities across Europe [417]. The three different data sources made available as part of the project were questionnaires, interview transcripts, and virtual reality (VR) simulations. The content of this data centers around learning disorders in higher education. Hence, the details in this section shall be heavily specific to this topic. Each source will be described to provide the context for the data modeling considerations in Section 3.2.

3.1.2 . Questionnaire

The Vrailexia project has collected valuable data from dyslexic and non-dyslexic students through questionnaires digitally distributed in high schools and universities in France and Spain.

Data Description The questionnaires capture the perception of students with respect to how potential difficulties affect them in their studies and how useful they would consider specific tools/strategies to cope with these challenges. Hence, the data collected from this source are purely personal, subjective opinions of the respondent. The questionnaires are provided in tabular form, serving as the first structured data source available for use. Table 3.1 describes the data source's structure and its main components. The questionnaire collects personal information related to the respondents, such as age, gender, dyslexic members in the family, and educational background. Furthermore, it aims to understand what are the respondents' potential learning disorders, learning difficulties and their perceived usefulness of tools and learning strategies.

Category	**Number of Columns**
Personal Information	45
Learning Disorders	6
Severity of Learning Difficulties (Scale 1-5)	13
Usefulness of Tools (Scale 1-5)	18
Usefulness of Learning Strategies (Scale 1-5)	22

Table 3.1: Breakdown of Questionnaire Columns.

Some of the tools and learning strategies are filled with the answer "I don't know" to indicate that a student was not familiar with a specific solution (see examples in Appendix table 3.6). There were a total of 2,106 respondents collected from both France and Spain. Approximately 23% of the respondents needed to be discarded as a result of leaving the majority of fields blank. 16% of the respondents had Dyslexia, often combined with other learning disorders. It proved difficult to collect data for a large percentage of dyslexic respondents, given that Dyslexia affects 5-17.5% of the population [339, 360]. The average age of respondents is 21.5, and the majority are Female (69.5%). The average rated severity across all problems by students with learning disorders is 3.16, compared to 2.43 for students without any learning disorder.

Data Pre-processing The pre-processing of the French and Spanish questionnaires involved several steps to clean and transform the data. The transformed columns were renamed to be more concise and descriptive. These final names would eventually be used as the names of the nodes modeled in the graph database. For example, questions such as "What is your age? Do not enter your date of birth" and "Which university are you from?" are reformulated to “Age” and “University” respectively. Some columns, such as the age, required some additional pre-processing as the answer formats were not consistent or were invalid. Overall, these pre-processing steps helped to clean and organize the questionnaire data, ensuring that it was in a suitable format for graph creation in Neo4j.

3.1.3 . Virtual Reality Simulations

As part of the present project, data collection from Virtual Reality (VR) simulations was performed with dyslexic students and non-dyslexic students. The purpose of the VR test is to investigate whether providing Dyslexic students with an interactive and immersive setting could enhance their learning experience, whilst also educating teachers on the considerations to make for students in such a condition [416].

Data Description Today, data has been collected in French, Spanish, and Italian universities. The participants are asked to perform two types of tests in a VR environment: 1) A Silent Reading test to assess performance; 2) A Psychometric Rosenberg [340] test for the assessment of anxiety, self-esteem and self-efficacy. The silent reading portion of the VR is a text comprehension exercise of which a respondent has to answer a series of elementary questions based on a text. The psychometric portion of the test (Rosenberg) seeks to survey the respondent’s level of confidence by asking them to rank a series of general questions on a scale of 1 (Strongly Disagree) to 5 (Strongly Agree).

The data is exported in tabular format in three separate tables, each storing the information about the user, the silent reading test, and the Rosenberg test, respectively. For example, the table storing the data regarding the silent-reading contains two columns for each of the six questions: the first column, a boolean representing whether the respondent answered the question correctly, and the second, the time that has elapsed (in seconds) since the beginning of the test upon the respondents completing their answer. A sample of a few columns from the three tables has been joined into one illustrative table in Appendix 3.7.

Overall, the data from the VR provides a complementary secondary structured data source with information about the respondents that would need to be integrated with data from the Questionnaire. At the time of conducting this research, only 100 responses were collected using the VR technology (of which 40%

Data Pre-processing As the first crucial pre-processing step, the names of respondents were anonymized by dropping the information for analysis. In the silent-reading test, response times were recorded in a cumulative manner each time a respondent answered a question. As the property of interest was the elapsed time for each individual question, the cumulative time records were transformed accordingly. In this test, the respondents' disorders were all collected in one column, and so the answers needed to be parsed such that each disorder was label-encoded. Finally, the age column required similar pre-processing as that described in the questionnaire by correcting answers provided in an invalid format.

3.1.4 . Interview Transcripts

The interviews' data was made available as text files of text-to-speech transcribed questions and answers with 10 French experts.

Data Description Various topics were covered related to students with learning disorders. Broadly speaking, the content of the interviews could be extracted and categorized into several themes: the learning disorders cited in each interview, problems encountered by students with such disorders, and finally, the tools/strategies that could be useful in coping with learning disorders or a specific difficulty. There were roughly 25 questions per interview. The interview transcripts serve as the unstructured data source in demonstrating the methodology.

Data Pre-processing No pre-processing was conducted on these files. However, the title of the page was removed, and the names of the experts and interviewers were anonymized by replacing them with "Expert" and "Interviewer" respectively.

3.2 . Methodology: an integrated graph Conceptual Schema

This section describes the methodological steps undertaken to model and integrate data from the multiple sources of interest. In Section 3.2.1, the modeling approach is introduced and later demonstrated with a conceptual schema of the structured and unstructured data sources. After that, Section 3.2.2 details the automation of data extraction. Section 3.2.3 addresses how the extracted entities were disambiguated. Finally, Section 3.2.4 describes the data resolution of instances originating from the different data sources.

The methodological steps of our study, as detailed in Section 3.2, are structured to enable the adaptation of our modeling approach to a broad array of use-cases. As mentioned previously in Section 3.1, and further justified in 3.5.1, Neo4j is the database system of choice to store information from the various data sources. Hence, the conception of the schema is conducted in a manner that follows the conventions of graph data modeling and Neo4j design. This means that schema representations are in property graph model form, whereas queries are demonstrated using Cypher: a query language optimized for property graphs [293].

3.2.1 . Modeling the Conceptual Schema

Conceptual models offer the ability to integrate heterogeneous sources, creating a base for uncovering insights and developing data-driven solutions. However, designing such conceptual models that deal with multiple sources can present multiple challenges. There are key differences in structure, format, and content between the sources as well as differences that may exist within each source itself. The following subsections 3.2.1 and 3.2.1 both describe our data modeling steps and introduce the pillars that compose the final schema of the integrated and interconnected graph database (Figure 3.6).

Structured Data Sources

As described in more details in Section 3.1.1, the questionnaire collects the following information about the respondents:

- their personal information
- their learning disorders, if any
- their self-assessment about how problems associated with the disorders affect them in their daily lives
- their perception of the usefulness of tools and learning strategies that are known to be used by students with learning disorders.

Given the central role of the respondents, we decided to model them as nodes containing their personal information as properties (e.g., anonymized identifier, age, and gender). Learning disorders (e.g., dyslexia, dysorthographia, dyspraxia, etc.) could also be treated as characteristics of respondents but were instead modeled as an independent node type, since there was an interest in capturing their relations to other nodes. Each problem, tool, and strategy was then categorized under their own respective node types as they interact with the respondent rather than being inherent characteristics. Under these modeling choices, each respondent was linked to the other four defined node types. Hence, the corresponding schema was centered around a dedicated node type called `Respondent`, as shown in Figure 3.1.

A `Respondent` node *HAS* a set of `Disorder` nodes and a set of `Problem` nodes. The `Respondent` node is also *HELPED_BY* sets of `Strategy` and `Tool` nodes. The relationships between the different nodes had to consider the answers of the respondent, who rated each problem, tool, and strategy on a scale from one to five, a measure of the respondent's connection with a specific node. These values were modeled as the Strength attribute of the relationship. As

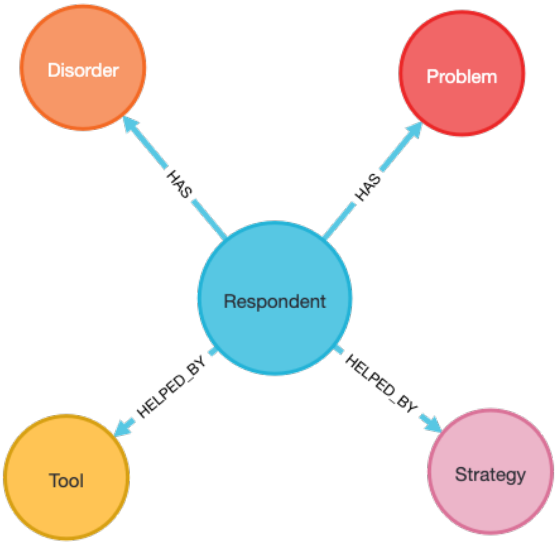


Figure 3.1: Conceptual schema of the relationship between the `Respondent` node and the nodes derived from the questionnaire.

an example, to find the respondents who consider certain problems to be the most severe, one could use a navigation scheme making use of the Strength edge attribute. In Cypher syntax:

```
(: Respondent)-[: HAS {strength: 5}]->(:Problem)
```

The answers of each respondent are easily traced back thanks to this representation. It was decided to relate every respondent to all the nodes of types `Problem`, `Tool`, and `Strategy`, irrespective of the strength of their answer. The one exception was in the case where the answer was left blank, as this meant that the respondent had no prior experience or knowledge about the concerned instance.

Storing the Strength attribute in the relationships instead of in the `Problem`, `Tool`, and `Strategy` (PST) nodes ensures that no information is lost and prevents node or attribute redundancies. This modeling choice further facilitates the use of graph science algorithms to process the database as a weighted graph. One limiting consequence, however, is that clustering algorithms, such as k-means, are restricted to node attributes in Neo4j [291]. In our schema, executing these functions would imply disaggregating edge properties (such as Strength), defeating the purpose of their modeled intent.

The VR data source, which serves as the second source of structured data, complements the questionnaire by providing further details about the characteristics of the respondents. The VR test is modeled in a similar way by creating relationships between `Respondent` nodes and the two additional node types (`Test` and `Confidence`), as illustrated in Figure 3.2. Answers from the silent reading test were modeled under Test nodes, while responses to the Rosenberg test fell under Confidence nodes. The two consequent relationships depict the cases where a respondent *ANSWERED* a test that measured their reading performance and *FEELS* a specific confidence level, as indicated through the Rosenberg questions. Similar to the Questionnaire, the test results and confidence level of the respondent are stored as an edge attribute. For example, if the goal was to identify respondents who answered a question correctly in less than ten seconds, the Cypher query for extracting the information from the graph is:

```
(: Respondent)-[r: ANSWERED {correct-answer: True})->(:Test)
WHERE r.time-taken < 10
```

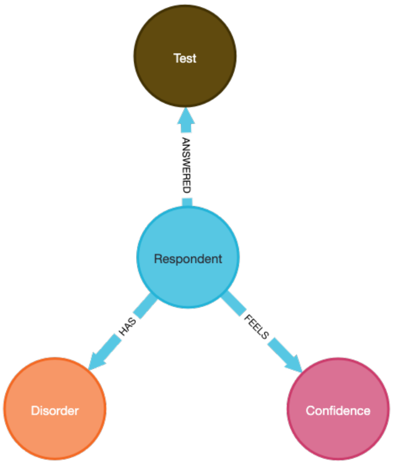


Figure 3.2: Conceptual schema of the relationship between a Respondent node with nodes derived from VR dataset.

The provided conceptual schema allows for a natural integration between the Questionnaire and VR test through the Respondent and Disorder nodes. In practice, it is important to consider that there are issues that require additional attention before achieving true integration. One such issue is the multilingual nature of the dataset (the questionnaire existed in both French and Spanish). Node names were stored in French by default but also had complementary attributes with machine translations in English, Spanish, and Italian. There is a long history of studies on the effectiveness of commercialized Neural Machine Translation Models such as Google Translate to translate in many languages across several applications [219, 425]. Another issue is to deal with cases where a respondent had contributed to both the questionnaire and VR tests. Some controls were implemented to address such cases to prevent any overwriting of personal information, thus ensuring proper and accurate data reconciliation. An additional attribute named Source was created within Respondent nodes to trace which data sources a respondent completed. Integrating these structured data sources was eventually rather straightforward thanks to the properties that were introduced for reconciliation. In contrast, the task was considerably more challenging for the unstructured data coming from the interviews.

Unstructured Data Sources

Modeling unstructured data sources is significantly less intuitive than that of their structured counterparts. Whereas structured data nodes and relationships can be intuitively interpreted, unstructured data sources require more complex considerations. Moreover, relying solely on human assessment could hinder any attempt to automate the data engineering pipeline. As described in Section 3.1.1, the unstructured data of this study was collected in the form of interview transcripts with experts to better understand the characteristics of learning disorders, the problems they may cause in higher education, and the ways in which affected students could address these problems. Using this information, it is possible to enrich the existing schema by modeling a new node type, `Expert`, that is critical for tracing the source of stored interview data. Each expert is modeled as a node with a unique anonymized identifier, having the language of the interview and the name of the transcript file stored as attributes of the node. Figure 3.3 highlights the schema modeled with this new node type.

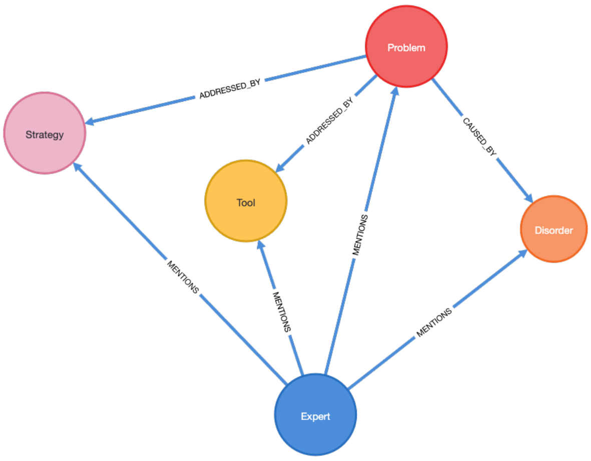
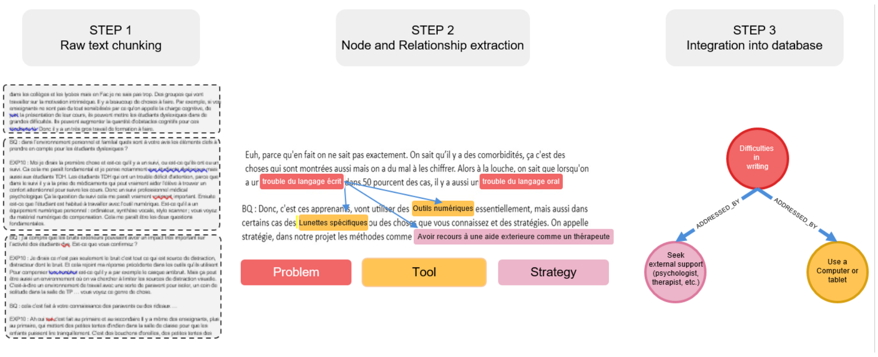


Figure 3.3: Conceptual schema of the relationship between the `Expert` node, the nodes derived from the interview, and the resulting causal relationships.

The `Expert` node as such *MENTIONS* other nodes. This enables both to trace the origin of any `Disorder` or `PST` nodes to their source expert. There are two additional relationships that can be further inferred from the interview data, namely that `Problem` nodes can be *CAUSED_BY* specific `Disorder` nodes and that `Problem` nodes can be *ADDRESSED_BY* `Tool` and `Strategy` nodes. These relationships are critical in that they create causal links between the different nodes, hence contributing to a more interconnected graph structure. Moving forward, a Named-Entity Recognition task was designed and implemented to efficiently extract data from the interview transcripts in an automated and scalable fashion. Its aim was to automatically extract information from transcripts according to the modeling decisions illustrated in Figure 3.3. The following subsection shall detail the methodology employed for this step of the data engineering pipeline.

3.2.2 . Data Extraction

As the structured sources are available in tabular form, categorizing the columns into their respective nodes is sufficient for loading data into the database. Specific transformations are made to facilitate the loading of such data into Neo4j, but these are not to be detailed as they are not the focus of this chapter. In contrast, the data of interest from the unstructured sources are not immediately accessible. In the example of the interviews, the data relating to each entity is scattered throughout the transcripts. Therefore, a Named Entity Recognition (NER) task is required before any database integration. An illustration of the task to be performed is proposed in Figure 3.4. To ensure the scalability of the data integration pipeline, it is imperative to rely on an automated method to conduct the extraction process. Our approach proposes to do so using OpenAI's "GPT-3.5-Turbo" Large Language Model (LLM). As previously discussed, this approach aims to demonstrate that with the correct data modeling choices and prompting, there is a promising path to automating data integration in a generalized manner without necessarily requiring heavy machine learning model training and deployment. The NER task conducted by the LLM needs to be able to perform node and relationship extraction like the one illustrated in Figure 3.4.



Euh, parce qu'en fait on ne sait pas exactement. On sait qu'il y a des comorbidités, ça c'est des choses qui sont montrées aussi mais on a du mal à les chiffrer. Alors à la louche, on sait que lorsqu'on a ur trouble du langage écrit dans 50 pourcent des cas, il y a aussi un trouble du langage oral

BQ : Donc, c'est ces apprenants, vont utiliser des Outils numériques essentiellement, mais aussi dans certains cas des Lunettes spécifiques ou des choses que vous connaissez et des stratégies. On appelle stratégie, dans notre projet les méthodes comme Avoir recours à une aide exterieure comme un thérapeute

Figure 3.4: NER-based integration of interview data.

As part of the process of extracting nodes and relationships, unstructured interview transcripts underwent a series of processing steps. First, these sources were segmented into manageable chunks to accommodate the LLM's context window - its input token limit. Each chunk was composed of a sequence of a question from the interviewer, followed by the corresponding expert's answer. Chunks were all assigned metadata containing their file name and exact location in the raw transcripts. As stated in Section 3.2.1, such information is later stored as node attributes to ensure the traceability of each piece of information.

Second, a prompt was constructed to employ the LLM to conduct the NER task. The drafted prompt provides details on the information to be extracted as well as formatting guidelines. Here, it was important to provide context in a manner that respected the schema previously shown in Figure 3.3. The prompt also incorporates a few-shot learning approach by feeding the LLM with example chunks along with the respective nodes and relationships that can be extracted from them. Constructing a strong prompt was particularly challenging, as the LLM can be prone to hallucinating or deviating from its specified task. There is no established comprehensive method yet to evaluate prompt design [5]. Hence, our prompt engineering step required many iterations and refinements to cope with the sensitivity of the LLM's interpretation of its provided instructions. The significant role of prompt optimization to improve results was also demonstrated in other studies [276]. An excerpt from our final prompt can be found in Appendix 3.8.

Third, a rigorous post-processing pipeline was implemented to ensure the proper formatting of the extracted entities. The LLM outputs were formatted strings of text, on which a series of controls were applied to ensure their conformity to the prompted instructions. Outputs were transformed into lists of nodes and relationships that were consequently loaded into the graph database. These entities were only introduced into the database if they respected the modeled schema in Figure 3.3. All imported node names were stored in their original language. Machine translations of these names were added as node attributes in all the other official languages of the Vrailexia project. This task was done as part of the NER process to ensure that the translations account for the context used by the LLM during extraction. New studies have already shown the competitiveness of LLMs at translation compared to traditional approaches [184, 426]. Our NER method enabled dealing with unstructured data in an automated way, the quality of which is further addressed in Section 3.4. Prior to that, a complementary task to NER in charge of handling extracted duplicates, called disambiguation, is described in the next subsection.

3.2.3 . Node Disambiguation

A disambiguation strategy was deployed as the final processing step of the unstructured data extraction to enhance data representation and trim out "near" duplicate node names from the NER task described in 3.2.2. The disambiguation involves computing textual embeddings of the node names and their pairwise cosine similarity values. The node names were first pre-processed to remove stop words and frequent words specific to each node type prior to computing these embeddings. Nodes with a cosine similarity of 0.98 and higher are flagged for merging. The duplicate candidates are consequently merged together by selecting one node name to be kept. Ideally, the preserved node name is the one with the most pairs in the duplicate groups. In cases where multiple nodes held the highest number of duplicates, the preserved name was randomly selected from among them. As this step aims to identify duplicates, it was reasonable, through trial and error, to set such a high threshold of cosine similarity.

Node disambiguation was not a focus of this work, but rather a sub-step between NER and entity resolution. The decision to further explore or optimize disambiguation in the future shall be made depending on the outcomes of these two steps. Nevertheless, disambiguation helped to ensure that the database does not suffer from a large volume of redundancies, which is essential to the data integration described in the next subsection.

3.2.4 . Data Integration and Resolution

The final step of the data engineering pipeline involves the integration of the dataset in a manner that enables navigation across multiple data sources. Specifically, data resolution (or entity matching/entity resolution) is achieved by connecting similar `Problem`, `Tool`, and `Strategy` nodes coming from heterogeneous sources. The illustrative example in Figure 3.5 depicts two `Problem` nodes coming from different sources conveying synonymous meanings. Data resolution aims to connect those two nodes. Such an operation was critical in our previous work [133], which aimed at developing a recommender system use-case based on a multi-source graph database. Data resolution was required to recover the insights from the expert interviews on how to address the most severe issues of Dyslexic students from the questionnaires. This task had, however, been previously handled manually in [133], requiring a considerable amount of time and representing an obstacle for automation.

Automating this approach faces a challenge: the inherent synonymy across the data sources is not always as explicit as illustrated in Figure 3.5. Similar nodes are often connected through analogous descriptions, contexts, or situations. Simply considering the cosine similarity of textual embeddings or resorting to other traditional semantic approaches is insufficient to capture such nuanced similarities [83] without introducing many false positives and false negatives. Therefore, it was necessary to take on the difficult endeavor of not only resolving nodes that had syntactic similarities as that shown in figure 3.5, but also resolving nodes having a contextual or nuanced common meaning, such as between “Reading Difficulties” and “Size of Text” (a relationship thematic in nature). An LLM prompting approach was again employed to systematically attempt the challenge of achieving data integration in an autonomous manner.

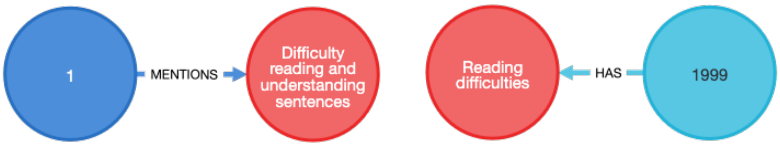


Figure 3.5: Example of two syntactically similar `Problem` node names originating from the questionnaire and interview transcripts respectively.

Several attempts were made at engineering a prompt that provided the LLM with sufficient context to label a pair of nodes. The final prompt defined that nodes would be linked if they shared one of three types of similarities: syntactic, thematic, or functional. Each similarity type was carefully defined in the prompt. In addition, the model was asked to explain the reason behind deeming a pair similar or not before providing a label. Research has shown that such chain-of-thought prompting could improve the ability of LLMs to conduct complex tasks [421]. The excerpt from our final prompt can be found in Appendix 3.9. An *IS_SIMILAR* relationship is introduced into the schema in cases where node names within the same node type are deemed similar and happen to originate from different data sources. The final conceptual schema, therefore, ensures interconnectedness, enabling comprehensive data analyses. Figure 3.6 shows the final schema, including the data resolution provided by this latest step.

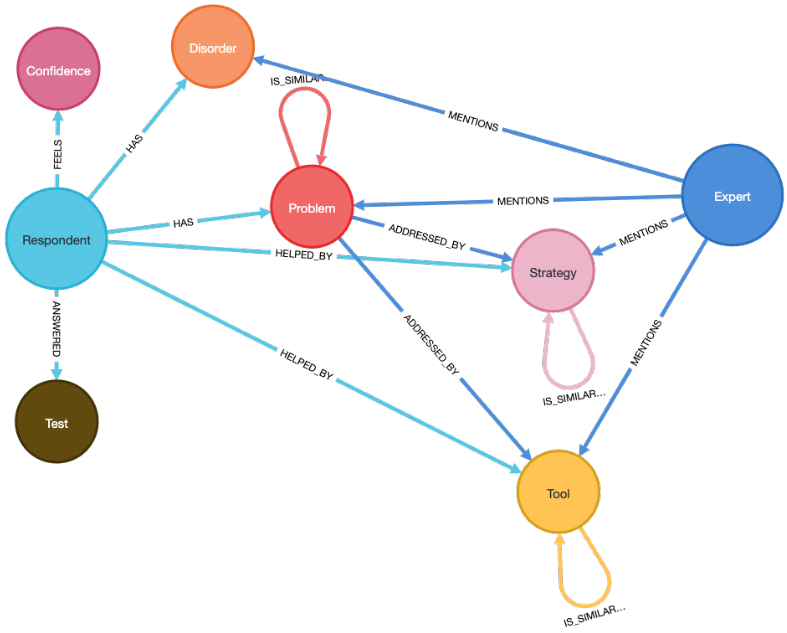


Figure 3.6: Final graph representation of the modeled schema of GRADD.

3.2.5 . Data for AI: The recommendation System

The unique structure of the GRADD schema offers opportunities for exploration. A recommendation system that can propose tools and strategies to dyslexic students based on their problems is a relevant use case that can demonstrate this. As part of this implementation, it was decided to focus only on the data originating from the interviews and questionnaires since no expert opinions relating to the VR test insights had yet been collected. The proposed recommendation system consists of two primary components, respectively in charge of:

1. Filtering candidate suggestions based on graph navigation by using causal links provided by experts.

2. Ranking suggestions by solving an ordinal classification task with a neural network.

The first building block in charge of filtering suggestions by graph navigation can be thought of as solving a link prediction problem. The link prediction task relies on the graph representation to extract valuable insights. By leveraging inputs from respondents and causal links provided by experts, subsets of relevant strategies and tools can be identified for specific problems. This process takes advantage of the inter-connectedness of the data in the graph structure, allowing for the extraction of meaningful relationships between various entities. This component of the recommendation system focuses on `Problem` nodes that a respondent has stated to be very severe (`Strength` edge attribute ≥ 4) to ensure the offering of targeted suggestions. Figure 3.7 illustrates an example of a respondent that has severe "Reading Difficulties" and for whom an expert recommendation is to propose, via a similarity link, "Audio recording of lessons" as a means to address this severe problem.

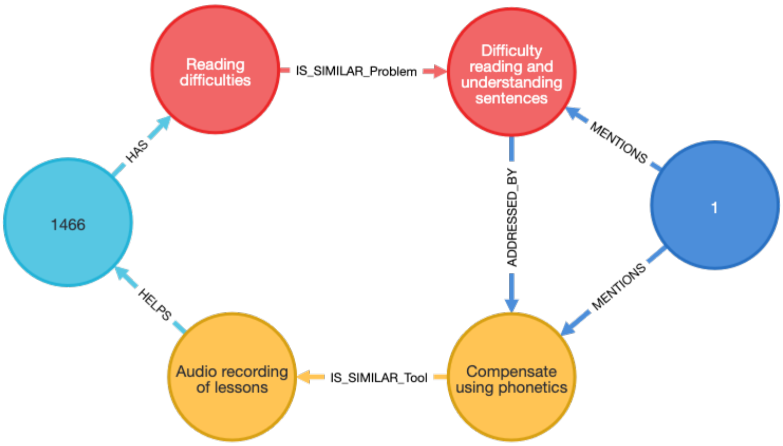

Reading difficulties
IS_SIMILAR_Problem
Difficulty reading and understanding sentences
HAS
MENTIONS
1466
ADDRESSED_BY
1
HELPS
MENTIONS
Audio recording of lessons
IS_SIMILAR_Tool
Compensate using phonetics

Figure 3.7: Graph navigation to retrieve causality link between problems and tools.

Using this approach makes it possible to filter lists of tailored suggestions for any respondent in the database, while capitalizing on the insights and connections made by experts in the field of dyslexia. This component also has the advantage of being extremely fast to execute, as it consists of a series of Cypher queries that can be run from a pipeline in Python or any other application with access to the database.

The second building block, the ranking algorithm, employs a neural network model to predict the usefulness that any respondent would assign to different strategies and tools on a scale of 1 to 5. The core objective of the neural network is, therefore, to solve an ordinal classification task. As inputs, the model uses various features extracted from the graph database. Once the ordinal classification task is performed, the predictions rank the sets of tools and strategies according to their inferred usefulness. Applying this ranking on the filtered suggestions provided by the first building block produces the final output of the proposed recommendation system - a tailored list of recommendations.

Both the model selection and the final set of input variables to be extracted from the graph database were determined based on performance on the test set. While more than 30 features were considered, it was finally concluded that only the respondents' disorders and problems would be selected, representing 17 input features. Considering the substantial number of target variables (about 40 tools and strategies), a neural network was employed for its ability to handle non-linear multi-output problems.

In terms of modeling, several architectures and loss functions were considered and compared. The model architecture was tuned to start from a 1-Layer network (with no activation function) up to a 4-Layer setup (with ReLU activations). It was decided to test and compare several appropriate loss functions as fully described in [133]. The Ordinal Log Loss (OLL) [323] [86], the self-guided EMD² loss (EMD²) [194], the CO2 loss [10] and the Soft-labels loss (SOFT) [66] have been implemented to address the problem of ordinality. The Mean Squared Error (MSE) and the Cross-Entropy (CE) were also employed for comparison, addressing the problem as regression and classification tasks, respectively. The model's predictive performance was evaluated using the ACC_1 (Accuracy within 1) [149], MAE, RMSE, and MMAE [109] metrics.

3.3 . Results and Discussion

In terms of evaluation, there was no quantitative way to benchmark the performance of the hybrid recommendations (graph navigation filtering + ranking algorithm) since they relied on the opinions of experts and would require feedback from dyslexic respondents. However, it was still possible to demonstrate that our data modeling has led to the successful ability to leverage the recommendations from experts using the graph navigation filtering. Furthermore, it is possible to quantitatively evaluate the ordinal classification algorithm used to rank these recommendations.

3.3.1 . Evaluation of the Ordinal Classification Algorithm

The final ordinal classification model was determined after comparing the performance of various architectures and losses. As part of the model selection process, hyperparameters such as the size of hidden layers, the learning rate of the Adam optimizer, and the number of epochs were tuned. The implemented losses also involved hyperparameters that were considered while tuning the models. Table 3.2 summarizes the best performance obtained per implemented loss function.

A critical detail is that the only evaluated outputs were those for which the usefulness was filled by a respondent since a blank answer would indicate they had not previously used the concerned tool or strategy. This minimizes the possibility of bias in the evaluation. During training, blank values of usefulness were replaced by the training set's sample means. This led to better performance results than when replacing blanks with zero, median, or mode.

Loss	Best Architecture	ACC_1	MAE	RMSE	MMAE
EMD²	3-Layer NN	**74.46 %**	**1.04**	1.34	1.81
OLL	3-Layer NN	74.13 %	**1.04**	**1.33**	1.80
MSE	2-Layer NN	72.69 %	1.07	1.37	1.83
CO2	2-Layer NN	68.90 %	1.14	1.54	2.70
SOFT	4-Layer NN	66.01 %	1.19	1.61	**1.75**
CE	3-Layer NN	65.53 %	1.18	1.58	1.76

Table 3.2: Best model per implemented loss function.

Among the various explored models, the OLL and the self-guided EMD^2 reached the highest ACC_1 of 74.13% and 74.46%, both with 3-Layer Neural Networks. As commented by the authors of [86], a value of 1.5 proved to be a good tradeoff for the OLL's hyperparameter α. The hyperparameters selected for the self-guided EMD² loss were $\lambda = 10^9$, $\omega = 1.5$ and $\mu = 0$.

Interestingly, setting $\lambda = 10^9$ amounts to discard the Cross-Entropy term. In [194], the EMD² term was introduced only as a regularizer to the Cross-Entropy loss because it faced convergence issues. These issues were also encountered experimentally by [86]. This observation was, however not met in our study, possibly due to the different nature of the data and the lower complexity in model architectures.

Both the self-guided EMD² and OLL losses showed significantly better performance than the other implemented losses on all metrics except MMAE. The best models on that metric were the Cross-Entropy (CE) and the Soft-labels (SOFT) losses. However, these models obtained significantly worse performances on all other metrics. While Cross-Entropy is known not to be the most appropriate choice for ordinal classification [10], the soft-labels relying on a label embedding designed explicitly for ordinal prediction tasks only had a slightly better performance. In light of these results, it was eventually decided to select the model using the self-guided EMD² loss as the ranking algorithm of the recommendation system.

3.3.2 . Demonstration of the Recommendation System Use case

In this section, a randomly selected dyslexic student is taken as an example to illustrate the results and corresponding discussion of using the hybrid recommendation system.

Most Severe Problems	Recommended tools
Reading Difficulties	1. Use a special font for easy reading 2. Use Audio Books 3. Numerical tutor (e.g., Siri) to which it is possible to query verbal explanations on challenging concepts 4. Words written in different colors
Difficulties to focus during online courses	1. A clearer presentation of the study material
Difficulties to understand complex or rare words	1. Register courses 2. Underline text with different colors 3. Conceptual sketches made by oneself 4. Repeat the studied contents 5. Summaries prepared by oneself

Table 3.3: Example of recommendations.

The first step would be to identify the respondent's most significant problems and utilize the graph navigation component of the recommendation system to use the experts' opinions and filter a list of relevant strategies and tools. In parallel, the trained ordinal classification model would take the respondent's declared disorders and problem strengths as inputs to infer the usefulness of all the strategies and tools. Combining both components by ranking the filtered suggestions in decreasing order of usefulness provides a tailored list of recommendations. Table 3.3 illustrates, as an example, the random respondent's three most severe problems and the corresponding top 5 tools to address each of them.

The results show that by navigating the graph, it is possible to use the knowledge of experts to recommend tools and strategies to address the most severe difficulties of dyslexic students who have answered the questionnaire. The technique is limited by the amount of expert interview data that exists. For example, the second problem displayed in Table 3.3 only has one recommendation because the current experts' opinions stored in the database only refer to this one tool as a method to address it. Hence, scaling the database to include more data will make such recommendations more refined.

3.4 . Discussion and Results

This section evaluates the performance of the proposed data extraction and integration pipeline. The feasibility of integrating such methods into the overall data engineering pipeline is assessed through the computation of common evaluation metrics. The section concludes with key takeaways and implications from this work.

3.4.1 . Named Entity Recognition

The NER conducted by the LLM loads a total of 1,011 `PST` and `Disorder` nodes, with the `Strategy` nodes forming the largest group of 360 distinct names (see full breakdown in Appendix 3.8). From these nodes, 345 relationships were imported into the database (see full breakdown in Appendix 3.9). Evaluating nodes and relationships generated by the LLM is challenging since the data sources are unstructured. The paragraph containing the exact location of entities is recorded to assess their actual relevance and validity. A common approach to evaluate a model's NER is to compute precision, recall, and the consequent F1-score [16, 76, 85, 248, 322].

A quality evaluation dataset was designed by sampling 10% of the raw chunks from the interview transcripts along with their respective generated nodes and relationships. We concede that such an approach can be prone to sampling bias. In fact, studies have considered this to be a demonstrative approach but have also noted the possibility of having a high variance in the results after sampling repetitions [256, 285]. Nevertheless, this demonstration could provide an understanding of whether further investigation into using such tools is worthwhile. The sample was stratified such that it represented content from all the experts. Three reviewers were then tasked to collectively read the sampled chunks and perform a manual NER to establish a ground truth of nodes and relationships. Their results were then compared to the ones' extracted from the model by recording the true positives (correctly identified node/relationship), false positives (falsely identified node/relationship), and false negatives (unidentified node/relationship). Table 3.4 below summarizes the results of the evaluation.

Entity	Recall	Precision	F1-score
All	75.41	69.84	72.49
Nodes	89.84	75.11	81.80
Relationships	52.87	58.64	55.34

Table 3.4: Sample quality evaluation of GPT-3.5-Turbo on NER Task (in %).

In terms of node extraction, an F1-score of 81.8% is high considering that the LLM has not been fine-tuned on this project's defined node types and rather simply given definitions with two corresponding examples. The reviewers noted that some of the sentences in the chunks were difficult to understand as a result of missing words or incorrect transcribing of speech-to-text. Data quality is surely a limitation that is difficult to improve without introducing extra steps that may limit the scalability of the pipeline. Looking further into the defects, an analysis of the false positives in the sample found that there are a few relevant examples that were assigned to the wrong node type. The `Disorder` nodes were the ones most affected by this issue. However, these nodes may hold a low impact on the overall database, as theoretically their weak semantic similarity to any of the nodes originating from the structured sources would lead them to have a very low graph degree. Other false positives were found to be due to node names composed of one word only, bearing no real meaning as a standalone. One such example of that was "Stubbornness", which was extracted by the LLM to be a `Problem` node. Such naming causes interpretation issues. One could wonder, for instance, if the problem refers to "dyslexics being stubborn", which would be completely wrong and misleading. After tracing back the chunk, "Stubbornness" was actually referring to the "stubbornness of teachers that sometimes refuse to accommodate the learning needs of Dyslexic students". As a consequence, a description attribute was later introduced as a takeaway from this issue: effectively backing up each node name with a contextual and detailed sentence. This description attribute was generated after completing the NER step by feeding the chunks again to the LLM, but this time with the extracted node names as context. In fact, this description attribute was integrated as a way to improve the quality of the Data Resolution task described in Section 3.4.2.

Since the relationships are extracted directly from the resulting nodes, the F1-score of 55.34% is unsurprisingly lower. An incorrect node classification automatically flags its relationships as false. Other research seems to find similar patterns in performance between nodes and relationships [85]. Therefore, the evaluation of extracted relationships should not be scrutinized with the same breadth. Interestingly, precision fared higher than recall. This lower recall was amplified by missing nodes from the node extraction task. The phenomenon was found to be especially true when the LLM failed to find `Disorder` nodes, which in turn caused the model to miss relationships with several distinct `Problem` nodes.

Overall, the results are very promising. The automatic pipeline was able to process all the interview transcripts and load the extracted information in about 1 hour, which could be made significantly shorter if task parallelization was introduced. In comparison, the human evaluation, comprising of three reviewers, working together to identify all the nodes and relationships for only a 10% sample, took 2.5 hours. Based on this, a naive estimate for a fully manual NER could be assumed to be about 25 hours. This is excluding the fatigue that could ensue over time and the breaks (in days) that could be required. Thus, the entity extraction by humans could easily take a few days for only 10 interview transcripts. The proposed automatic pipeline, therefore, surely offers strong potential gains in terms of time consumption.

While the current approach already offers encouraging results, there are several avenues that can be explored to improve the NER task's F1-score. An LLM could be fine-tuned to learn the nodes and relationships in a domain-specific way. This would require creating a ground truth and also accept that the model would be specialized on a certain corpus of nodes and relationships [53]. Another potential solution is retrieval augmented generation (RAG) [53], a method that enriches the context provided to the LLM using a knowledge base. In fact, RAG can be further enhanced by following a framework [385]. Finally, the prompt generation can be delegated to a secondary LLM that has been fine-tuned to generate instructions specific to NER [275, 420]. The benefits of such potential improvements can apply to a wide array of tasks, including the one covered in Section 3.4.2.

3.4.2 . Data Resolution

The data resolution task can be thought of as a binary classification task. Hence, the same metrics of Section 3.4.1 shall be used. There were 17,981 potential similar pair of nodes extracted from the questionnaires and interviews. In a real-world setting, it would not be practical to evaluate the total set and so 2% of the pairs were evaluated. Two reviewers were tasked to determine whether a pair of nodes was similar by simply labeling 1 when similar, or 0 otherwise. The reviewers were provided the same instructions as the LLM to define the context of when to classify a pair of phrases as similar. They were also provided node descriptions to help understand the context of nodes extracted from the interviews, as described in Section 3.4.1. Finally, the reviewers were privy to the node type of each assessed pair. This information was not provided to the LLM to prevent entity matching biases, potentially induced by the pair sharing the same node type.

The sampling strategy was meticulously designed to ensure an equal distribution between positive and negative instances to diligently evaluate the data resolution task. As the dataset was significantly unbalanced (thought to have less than 10% of the positive examples), a special method was adopted to streamline the sample creation. The 17,981 pairs were sorted in descending order of pairwise cosine similarity to increase the likelihood of sampling positive examples. The group of reviewers consequently determined whether a pair was similar until 1% of the total number of pairs was filled with positive examples. Evidently, as a result of the previously mentioned imbalance in the dataset, an equal number of negative examples were also identified through this iterative procedure. The fact that all these negative examples were sourced from the pool of high cosine similarity indicates that it is more challenging for the LLM to avoid false positives compared to resorting to random sampling.

Table 3.5 outlines the results of the LLM on the data resolution task for the selected sample on GRADD. The results of the LLM were benchmarked against a baseline model that clustered the node names' textual embeddings using OpenAI's "ada-002" model. This baseline approach aims at grouping similar nodes together. It effectively identifies synonymous entities originating from different data sources, categorizing them under common cluster identifiers. The clustering was conducted using k-means, assigning the optimal value of k based on the highest average silhouette score.

Model	Node Type	Precision	Recall	F1-score
Baseline: Clustered Embeddings	Problem	**91.67**	29.72	44.40
	Tool	36.00	24.32	23.03
	Strategy	**85.71**	16.22	27.27
	Total	59.09	23.42	33.54
GPT-3.5-Turbo	Problem	63.83	**81.08**	**71.42**
	Tool	**63.33**	**51.35**	**56.72**
	Strategy	80.77	**56.76**	**66.67**
	Total	**67.96**	**63.06**	**65.42**

Table 3.5: Results of the Data Resolution Task (in %).

The LLM outperformed the baseline on all metrics when looking simply at the "Total" values, achieving a final F1-score of 65.42%. The Baseline outperformed only on the precision metric of the `Problem` and `Strategy` nodes. Relying on textual embeddings, the Baseline model reached high precision by simply finding most of the syntactic similarities, such as "Reading Difficulties" and "Difficulty to Read". However, as illustrated by its poor recall, this model is unable to satisfy the requirements for contextual and thematic similarities, such as between "Reading Difficulties" and "Size of Text" or between "Text with every other line highlighted" and "use colors to underline text". This is interesting considering that we expected that the LLM would be at a disadvantage as a result of our sampling strategy biasing toward higher cosine similarity, hypothesized to benefit the clustering of textual embeddings. The higher precision for `Problem` and `Strategy` nodes is therefore explained by the model only classifying a pair as similar in a very small portion of instances, limiting the chances of causing false positives. It is somewhat surprising that the precision of the Baseline on `Tool` was very low. Upon investigation of the examples, it was found that many of the unrelated pairs of `Tool` names contained the word "Dyslexic" or "Dyslexia", increasing their cosine similarity and misleading the baseline model to generate false positives. Considering this, it is impressive that the LLM was able to cope with such pitfalls and classify correctly such nuanced examples as those provided above. It is worth noting however that in a considerable number of false positive examples, the model was providing too broad justifications for thematic similarity. For example, a pair of `Problem` nodes were labeled as similar because they were both "describing a difficulty in an educational setting" - the definition of the `Problem` node type. Ironically, a prompt optimization that attempted to correct this by giving the model context about the pair's node type yielded a slightly lower precision.

In addition to the potential improvements proposed in Section 3.4.1, one can simply use a more advanced model like GPT-4, which has been shown to yield higher F1-scores at entity resolution [316]. Moreover, one can change the prompt to only focus on syntactic similarities if one faces a use-case that does not require such implicit definition of similarity for data integration. However, it could be more interesting to change the conception of the data modeling to accommodate for the fact that language is in reality nuanced and that not all relationships are simply syntactic in nature. For example, the prompt can be modified to also provide a confidence score if a pair is deemed similar [247]. Even though some studies observed that such method yielded case-dependent results [365], this probability could be stored as an edge attribute of the similarity link, allowing for a more in-depth analysis within the graph database. Alternatively, the modeling of the relationship, *IS_SIMILAR*, can be modified to allow for three different possible relationships between two nodes from different data sources. For example, the relationships could be *IS_SYNTACTIC_SIMILAR*, *IS_THEMATIC_SIMILAR*, and *IS_FUNCTION_SIMILAR*: the three possible contextual similarities defined to the LLM, as mentioned in Section 3.2.4. This further exemplifies the importance of conceptual data modeling. In fact, [287] has constructed a semantic framework to help human experts define a more comprehensive strategy to dealing with similarities when attempting to integrate heterogeneous data sources. Combining such frameworks with our explored methodology may enhance the semantic capabilities of LLMs.

3.4.3 . Key Takeaways and Implications

To summarize, this research did not aim to find the perfect automatic tool for data integration, but to explore the potential of Large Language Models (LLMs) in enhancing this process. The findings reveal that LLMs hold great promise. Minor adjustments to prompts significantly impacted F1-score (increased by a factor of 1.76 in the case of entity resolution), highlighting the sensitivity of these models. Data modeling proved invaluable for crafting effective prompts and contextualizing the instructions, reinforcing the idea that while technology aids, it cannot replace the foundational task of data modeling. Post-processing the LLM's output emerged as a critical step, addressing issues like formatting errors, token limits, and incorrect node or relationship generation. This underscores the importance of a robust data integration pipeline to manage such challenges, indicating areas for further refinement and exploration in the realm of data privacy and processing efficiency.

Acknowledging the limitations of our work is equally important for a comprehensive understanding. The work has shown that the data engineering pipeline can be automated in a manner that aids humans. However, it is still unclear how such approach can be scaled to big data applications. The computational complexity of the tasks, especially that of entity resolution, could pose a problem in cases of high volumes. Regardless, such methodology can prove to be vital to practitioners not operating in such cases. Another limitation stems from the inherent bias associated with using a sample evaluation. This does not diminish the conclusions themselves, however, it is important to work and establish a comprehensive sampling framework to evaluate such large datasets, considering that a full evaluation is probably unrealistic in most use-cases. Finally, the data privacy concerns relating to using LLMs cannot be ignored. If such concerns arise, one could rely on open-source models, such as Mixtral [214], Mistral [213] or Llama2 [393], if the right resources are available.

3.5 . Related Work

Our approach lies in creating a graph representation of data coming from different sources to enable the execution of predictive Artificial Intelligence algorithms [133]. Achieving this objective requires appropriate data engineering considerations, including the definition of a conceptual model to help design, develop and run these artificial intelligence solutions [260, 263, 395]. New research fields are opening strong opportunities for the definition of conceptual models [45, 97]. Simultaneously, research has also been devoted to addressing data integration with novel approaches [44, 168, 285]. As a result, it is imperative to investigate advancements in both fields: modeling and integration. Prior to that, since the implementation is performed in Neo4j, a portion of this section is dedicated to further understanding the benefits of using such a tool.

3.5.1 . Data Modeling using Neo4j Graphs

One of the key challenges of our study has been to integrate various sources of information of different structures and languages. For example, [366] already considered that integrating diverse and complex information such as structured databases, unstructured text, and multimedia content represented a significant challenge in Big Data applications. NoSQL databases have been discussed as an appropriate solution for such endeavors due to their ability to adapt to different sources and data formats, as well as their high-performance capabilities and enhanced flexibility [348].

Graph data structures, which belong to the NoSQL family, are applied in areas where information about data inter-connectivity or topology is of great importance [25]. Modeling data as graphs allows querying relationships in the same manner as querying the data itself. Instead of calculating and querying the connection steps as in relational databases, graph databases read the relationship from storage directly [25]. Neo4j employs the so-called Property Graph Model [26]. Like any other graph database model, it relies on two types of entities: nodes and edges. However, Property Graphs contrast with other graph data models in the way that they allow the storing of properties directly on nodes and edges [26], which is not the case for other graph data models such as RDF [354]. Recent literature [289] commented on how graph databases are easily scalable, fast, efficient, and flexible. This was confirmed by [88], which explores time-evolving social network modeling achieved through utilizing Neo4j. The objective was to capture human activities and interactions sourced from mobile devices and wearable sensors. Notably, the study showcases the effectiveness and scalability of real-world queries, highlighting the efficiency of the approach [88]. Our study capitalizes on the capabilities of Neo4j to establish a directed graph, facilitating the visualization of pertinent insights. The choice of Neo4j was particularly interesting, as it offered us the abilities to take advantage of the interconnectedness of a graph structure, while handling different data sources in a flexible and integrated manner.

3.5.2 . Conceptual Modeling and Artificial Intelligence

Datasets are nowadays analyzed by algorithms and systems with growing complexity. Conceptual modeling has always been instrumental in understanding data and complex systems. For decades, the research community has dedicated large attention to modeling and dug in topics that include data modeling, process modeling, meta modeling, and model quality [46, 187, 382]. One of the main questions during the last few years has been: "how conceptual modeling can help structure machine learning and practitioners' projects?" [115, 263, 432]. The conclusion has been that machine learning and data modeling can complement and help each other [147] even to the point of defining systems that can auto-configure and optimize themselves [253]. The attention around this topic is increasing to the point that a new research area identified with the CMAI acronym (Conceptual Model and Artificial Intelligence) recently started to be developed [73]. In a similar vein, our conceptual modeling work has been oriented to complement value adding AI applications, such as the recommender system proposed in our previous study [133]. Our approach adopts a reciprocal approach by taking advantage of Large Language Models to enhance data engineering tasks.

3.5.3 . Advances in Data Modeling and Integration

Defining a good conceptual model is still an open challenge in many research areas. Even recent literature shows how a big research community is still working on defining and validating conceptual models for use-cases such as smart homes [424], European laws [345] or even manufacturing business analytics [297]. Similarly, studies have shown that defining a conceptual model that integrates many heterogeneous data sources is an even more complex and open challenge [14, 382]. Many open questions persist, particularly in the context of new tools and approaches like LLMs [168, 285] or the synergy between knowledge graphs and natural language [222].

The last several years have seen significant efforts to explore the use of Natural Language Processing (NLP) techniques and applications of language models in the context of databases systems and conceptual modeling [160, 331, 396]. These applications also include data discovery and integration [226, 285, 403]. For example, very encouraging results have emerged in using GPT-3.5 for the task of entity extraction from unstructured documents [43, 85]. Other works such as [45] and [97] propose LLM-based tools that extract document values from data lakes. Recent research has also focused on considering GPT-3 in support of model construction and definition [397] or data transformation [358]. Some studies even attempted to substitute databases and data models with Generative AI Machines [218].

The problem of data integration has been widely studied in literature [44, 64]. Classical solutions traditionally define a unified framework based on general meta-structures and a set of rules to map the sources into a target model [154, 170]. In a similar fashion, our work maps all the available data into a target schema made of entities coming from different data sources. According to our research, a conceptual model is indeed essential to succeed in integrating data from heterogeneous sources. That is why, our present study explores how LLMs can be used to support the automation and enrichment of a graph data model. This research field is only starting to be explored, but some approaches have already shown good results [43, 45, 97, 397].

3.6 . Conclusion

This study demonstrates the effectiveness of a novel application of Large Language Models (LLMs) for integrating heterogeneous data sources into a graph database. Through a comprehensive methodology that includes data modeling, extraction, and integration, supported by technologies such as Neo4j and GPT-3.5-Turbo, complex data processing tasks can potentially be streamlined. Although the data modeling choices have been centered around one specific dataset, several steps such as those relating to the modeling of entities as well as the decision of where to store attributes can be expanded to other use-cases, especially in the context of an educational environment. The evaluation of both Named Entity Recognition and Data Resolution tasks illustrates the effectiveness and efficiency of LLMs in handling diverse data types. The project highlights the synergy between human expertise in data curation and AI's capabilities: opening avenues for more nuanced and scalable research databases.

Our future work aims to develop a more robust framework for data modeling that can better capture the complexities of educational data. The development of such a framework could also include an exploration to enhance an LLM's understanding of nodes and relationships by leveraging techniques such as retrieval augmented generation (RAG) and further prompt engineering. Furthermore, data modeling can be improved by accounting for the nuanced nature of language, potentially employing probabilistic approaches to similarity and exploring the inclusion of syntactic, thematic, and functional relationships into the conceptual schema. Moreover, since model fine-tuning is difficult due the lack of available ground truth, it is worthwhile investigating generating a synthetic dataset using LLMs that are specifically tailored to the use-case [388]. It has also been established that different results could be obtained from repeated executions of LLMs [63]. To assess the robustness of the proposed approach, it could be interesting to perform a statistical analysis on multiple runs of the data extraction and integration processes. This quantitative evaluation could also provide the opportunity to compare the robustness of different language models on this specific task. Finally, the optimization of the disambiguation process presents a rich avenue for further research that is not covered here, as this study primarily focused on data extraction and resolution.

3.A . Nature of Data Used

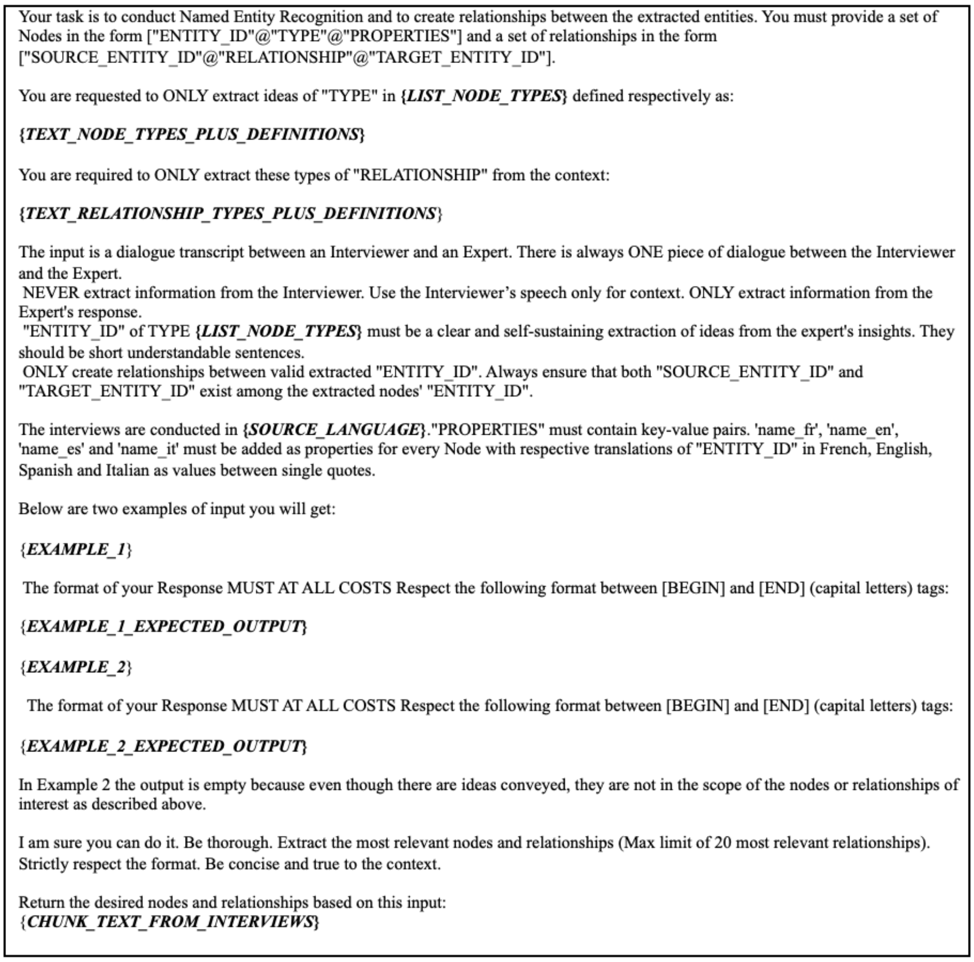
Variable	**Respondents**	
id	155	34
How old are you? (do not enter your date of birth)	22	229
Gender	F	Prefer Not To Say
Dyslexics in Family	Mother, Brother	
What university are you from?	Nanterre Univ.	CentraleSupélec
Are you dyslexic?	Yes	No
Have you been diagnosed with dyslexia?	Yes	
***IF YOU ANSWER YES TO THE PREVIOUS QUESTION* - What other difficulty(s) do you have besides dyslexia? [Calculation difficulty - dyscalculia]**	Yes	
***IF YOU ANSWER YES TO THE PREVIOUS QUESTION* - What other difficulty(s) do you have besides dyslexia? [Other]**		
Reading Difficulties	5	2
Presentation Attention	4	4
Audiobook Quality	I don't know	2
Images for Words	4	2
Oral Exams	3	1

Table 3.6: Example Data from Select Columns of the Questionnaire.

Variable	**Respondents**	
id	361	362
created_at	2022-12-12 10:56	2022-12-12 18:00
age	22	32
sex	female	male
dyslexia_type	Dysorthography	Dyscalculia
language	4	4
“Press quickly and twice in a row the yellow button” Time	81.0019	44.9134
“Press quickly and twice in a row the yellow button” Correct	TRUE	TRUE
“Try to say the word kiss/bisous/beso/bacio” Time	0	64.3253
“Try to say the word kiss/bisous/beso/bacio” Correct	FALSE	TRUE
“I feel that I am a person of worth, at least on an equal plane with others”	1	2
“I feel that I have a number of good qualities”	1	2

Table 3.7: Example Data from Select Columns of the VR Set.

3.B . Sample Prompts for Data Integration



Your task is to conduct Named Entity Recognition and to create relationships between the extracted entities. You must provide a set of Nodes in the form ["ENTITY_ID"@"TYPE"@"PROPERTIES"] and a set of relationships in the form ["SOURCE_ENTITY_ID"@"RELATIONSHIP"@"TARGET_ENTITY_ID"].

You are requested to ONLY extract ideas of "TYPE" in **{*LIST_NODE_TYPES*}** defined respectively as:

{*TEXT_NODE_TYPES_PLUS_DEFINITIONS*}

You are required to ONLY extract these types of "RELATIONSHIP" from the context:

{*TEXT_RELATIONSHIP_TYPES_PLUS_DEFINITIONS*}

The input is a dialogue transcript between an Interviewer and an Expert. There is always ONE piece of dialogue between the Interviewer and the Expert.
NEVER extract information from the Interviewer. Use the Interviewer's speech only for context. ONLY extract information from the Expert's response.
"ENTITY_ID" of TYPE **{*LIST_NODE_TYPES*}** must be a clear and self-sustaining extraction of ideas from the expert's insights. They should be short understandable sentences.
ONLY create relationships between valid extracted "ENTITY_ID". Always ensure that both "SOURCE_ENTITY_ID" and "TARGET_ENTITY_ID" exist among the extracted nodes' "ENTITY_ID".

The interviews are conducted in **{*SOURCE_LANGUAGE*}**."PROPERTIES" must contain key-value pairs. 'name_fr', 'name_en', 'name_es' and 'name_it' must be added as properties for every Node with respective translations of "ENTITY_ID" in French, English, Spanish and Italian as values between single quotes.

Below are two examples of input you will get:

{***EXAMPLE_1***}

The format of your Response MUST AT ALL COSTS Respect the following format between [BEGIN] and [END] (capital letters) tags:

{***EXAMPLE_1_EXPECTED_OUTPUT***}

{***EXAMPLE_2***}

The format of your Response MUST AT ALL COSTS Respect the following format between [BEGIN] and [END] (capital letters) tags:

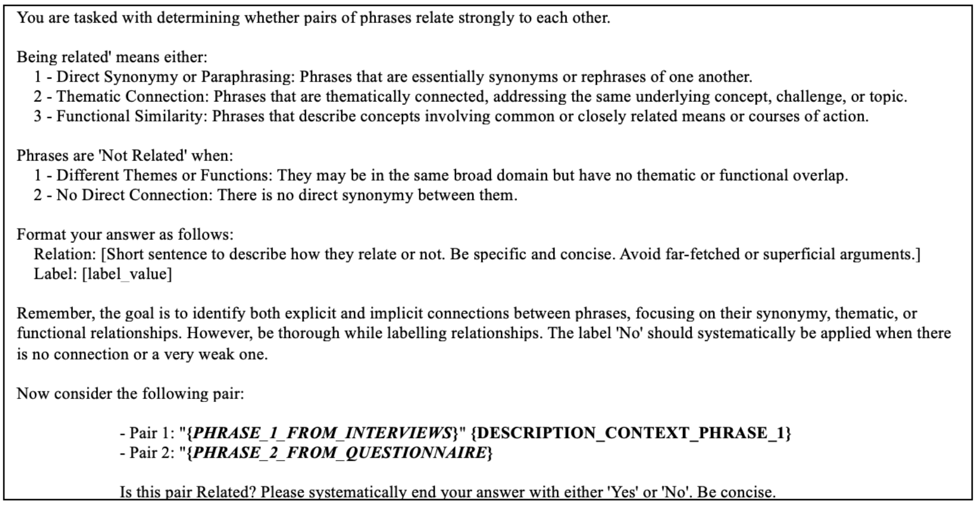
{***EXAMPLE_2_EXPECTED_OUTPUT***}

In Example 2 the output is empty because even though there are ideas conveyed, they are not in the scope of the nodes or relationships of interest as described above.

I am sure you can do it. Be thorough. Extract the most relevant nodes and relationships (Max limit of 20 most relevant relationships). Strictly respect the format. Be concise and true to the context.

Return the desired nodes and relationships based on this input:
{***CHUNK_TEXT_FROM_INTERVIEWS***}

Figure 3.8: Excerpt from prompt used for NER task (text formatted as ***TEXT*** are user inputs).



You are tasked with determining whether pairs of phrases relate strongly to each other.

Being related' means either:
1 - Direct Synonymy or Paraphrasing: Phrases that are essentially synonyms or rephrases of one another.
2 - Thematic Connection: Phrases that are thematically connected, addressing the same underlying concept, challenge, or topic.
3 - Functional Similarity: Phrases that describe concepts involving common or closely related means or courses of action.

Phrases are 'Not Related' when:
1 - Different Themes or Functions: They may be in the same broad domain but have no thematic or functional overlap.
2 - No Direct Connection: There is no direct synonymy between them.

Format your answer as follows:
Relation: [Short sentence to describe how they relate or not. Be specific and concise. Avoid far-fetched or superficial arguments.]
Label: [label_value]

Remember, the goal is to identify both explicit and implicit connections between phrases, focusing on their synonymy, thematic, or functional relationships. However, be thorough while labelling relationships. The label 'No' should systematically be applied when there is no connection or a very weak one.

Now consider the following pair:

- Pair 1: "***{PHRASE_1_FROM_INTERVIEWS}***" **{DESCRIPTION_CONTEXT_PHRASE_1}**
- Pair 2: "***{PHRASE_2_FROM_QUESTIONNAIRE}***

Is this pair Related? Please systematically end your answer with either 'Yes' or 'No'. Be concise.

Figure 3.9: Excerpt from prompt used for Entity Resolution task (text formatted as ***TEXT*** are user inputs).

3.C . Summary of Data Extracted from Expert Interviews Using LLM

Entity	**Number of Nodes**
Disorder	61
Problem	314
Tool	276
Strategy	360

Table 3.8: Extracted Nodes.

Entity Pair	**Number of Relationships**
(Problem, Disorder)	83
(Problem, Tool)	125
(Problem, Strategy)	137

Table 3.9: Extracted Relationships.

4 - Data integration in smart cities and energy conversion

Cities serve as vital hubs of economic activity and knowledge generation and dissemination. As such, cities bear a significant responsibility to uphold environmental protection measures while promoting the welfare and living comfort of their residents. There are diverse views on the development of smart cities, from integrating Information and Communication Technologies into urban environments for better operational decisions to supporting sustainability, wealth, and comfort of people. However, for all these cases, data is the key ingredient and enabler for the vision and realization of smart cities.
One key enabler of cities and smart cities is energy. During the last years' energy transformation, often referred to as energy conversion, green hydrogen technologies and efficiencies are critical components of the plan to achieve net-zero CO_2 emissions. Thus, the use of artificial intelligence (AI) and machine learning (ML) tools in these fields could pose opportunities to accelerate and optimize the performance and efficiencies of energy conversion tasks. For this task, we conduct a study about the use and acquisition of real experimental data, over simulated data, and overall standardized explicit analysis of the data size, accuracy, or error rates achieved, and comparison of the performance of algorithms with a benchmark.

The chapter is adapted from the following papers:

- Ekaterina Gilman, Francesca Bugiotti, & all *Addressing Data Challenges to Drive the Transformation of Smart Cities*. ACM Trans. Intell. Syst. Technol. 2024
- Konstantinos Mira, Francesca Bugiotti, Tatiana Morosuk *Artificial Intelligence and Machine Learning in Energy Conversion and Management*, Energies 2023

4.1 Background and context 68
4.2 Data challenges in the context of smart cities 84
4.3 Related work 108
4.4 Conclusion 111

This chapter explores the challenges associated with smart city data from a data integration point of view. We start with gaining an understanding of the concept of a smart city, how to measure whether the city is a smart one, and what architectures and platforms exist to develop one. Afterwards, we research the challenges associated with the data of the cities, focusing on availability, heterogeneity, management, and analysis. Lastly, we analyze how effective data analysis can improve the energy conversion field as it is a key enabler for smart cities.

4.1 . Background and context

Cities play a crucial role as the engines of the economy and centres of connectivity, knowledge, and services [402]. Based on the estimation from the United Nations, 66% of the world's population will live in urban areas by 2050 [288]. Being the centres of growth and innovation, cities need to address significant challenges for environmental protection and citizens' prosperity and living comfort. These challenges become pronounced in large and rapidly growing cities, which concurrently struggle to establish robust infrastructure to ensure clean air and water, energy supply, food, transportation, efficient waste management, and provisioning of public spaces - vital components for human well-being [241].

Cities are increasingly equipped with Information and Communication Technology (ICT) technologies to improve their resourcing and the quality of life of their inhabitants, ultimately becoming smart cities. The term "smart sustainable city" is used to denote a city that is supported by the widespread adoption and extensive use of advanced ICT, which, coupled with various urban systems and domains and strategic coordination of their intricate interrelations, empowers the city to manage available resources sustainably and efficiently for improved economic and societal outcomes [70]. Cities are becoming smart and sustainable in ways that enable us to monitor, understand, analyse and plan the city to improve the efficiency, equity, and quality of life for citizens in real time [59].

Smart cities are technologically modern urban areas leveraging networked systems to collect data and data analytics platforms to analyze data. The development of smart cities requires the integration of various subsystems to work together to achieve a common goal, which is a system of systems approach. A system of systems is a collection of independent but interrelated systems that are developed and operated to meet a common set of objectives. In the context of smart cities, a system of systems integrates multiple subsystems, such as transportation, energy, water, waste management, and public safety, into a single system. Such integration is crucial to achieve the common goal of improving the quality of life for citizens. In order to integrate these subsystems, smart cities rely on data [89].

Data is the key ingredient and enabler for the vision and realization of smart cities. A huge volume of data represents a large amount of information generated via and about people, objects, and interactions among them in smart cities. Such data produced in different sectors within a city can contribute in generating useful information for various stakeholders for decision making, such as policy makers, citizens, domestic governance bodies, and industrial communities [148]. By analyzing data in smart cities, we can potentially understand the activities and interactions and enhance the quality of the services offered to the citizens, as well as provide benefits for city management, like contributing to lowering operational expenses. For example, in Seoul, the government has been collecting data related to healthcare, transportation, and residency to make it available to citizens and scientists [252]. From these data, various smart services can be developed leveraging ICT and big data solutions [7, 69, 96, 179, 273, 350]. However, there are many challenges that need to be tackled on the way from the "raw" data to the smart service, from data and system perspectives.

Smart city data integration and analytics platform is responsible for integrating data from various sources into a single system and performing analytics. The platform uses data integration tools and techniques to extract, transform, and load data from various sources, such as databases, sensing systems, and other monitoring devices. Once the data is integrated, it can be analyzed to provide insights into various aspects of city life. For example, data from traffic sensors can be used to optimize traffic flow, reduce congestion, and improve public transportation. Similarly, data from energy consumption meters can be used to optimize energy usage, reduce costs, and improve energy efficiency. In order to manage all of this data, smart cities rely on a variety of data management systems, which are responsible for analyzing the integrated data to provide insights into various aspects of city life. The systems typically use a variety of data analytics techniques, such as machine learning and artificial intelligence, to analyze the data and identify patterns and trends. These insights can be used to optimize various aspects of city life, such as traffic flow, energy consumption, and waste management.

This survey provides a holistic view covering data-related challenges of smart cities, see Figure 4.1. We start by defining and measuring smart cities and survey recent works of smart city architectures and platforms. Having this understanding, we then dive into exploring challenges and solutions for handling data. Given the diversity of smart city data, we also review challenges related to data heterogeneity and integration. Then, we dive into data management issues, including data acquisition, storage, processing, and governance. After that, we explore challenges related to data analysis, ethics, data privacy, and security.

This chapter contributes to a careful investigation of challenges associated with the data in smart cities. In a nutshell, our contributions are twofold:

1. Providing a comprehensive review of the latest development of the smart

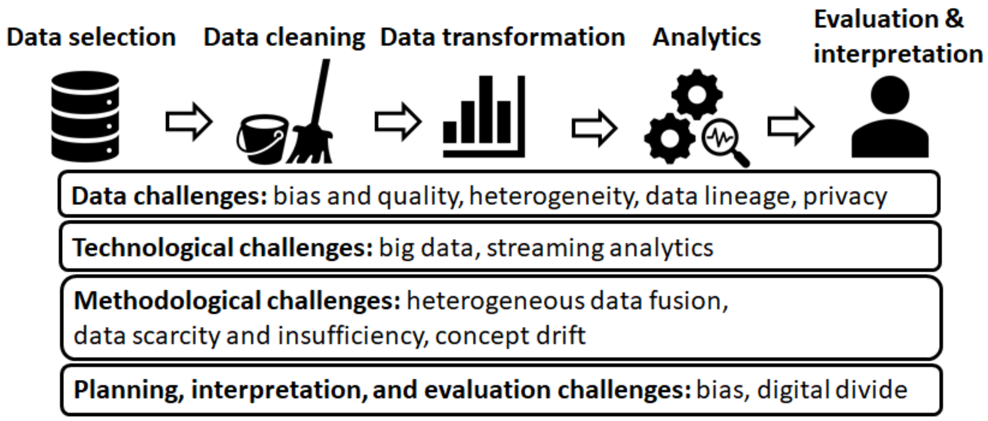

Data selection
Data cleaning
Data transformation
Analytics
Evaluation & interpretation
Data challenges: bias and quality, heterogeneity, data lineage, privacy
Technological challenges: big data, streaming analytics
Methodological challenges: heterogeneous data fusion, data scarcity and insufficiency, concept drift
Planning, interpretation, and evaluation challenges: bias, digital divide

Figure 4.1: Topics covered in the manuscript.

city concept. We also review existing solutions allowing for measuring smart cities as well as architectures and platforms for developing smart cities.

2. Exploring challenges associated with the data of the smart cities, covering data availability and quality, heterogeneity, management, analysis, privacy, security, and ethical aspects, and a research agenda for addressing these challenges.

This work aims to serve as a "one-stop shop" covering data-related issues of smart cities with references for diving deeper into particular topics of interest. The remainder of this chapter is organised as follows. As first, we discuss several significant research challenges, such as data availability, quality, heterogeneity, management, analysis in Section 4.2.
Section 4.3 summarises related work. We finally conclude by presenting a detailed discussion in Section 4.4.

4.1.1 . Defining smart city

The smart city concept is flexible and open, which is probably a central factor behind its popularity and global success. At the same time, it is also notoriously challenging to define [290]. The reasons are two-fold. On one hand, scholars have mapped and categorized smart city development in different ways, depending on their background [231, 280]. On the other hand, different cities around the world have applied the agenda in their own terms, due to their specific economic, political, legal, social, and cultural arrangements [9]. Figure 4.2 presents a high-level evolution of smart city concept development. In general, the smart city refers to optimizing city processes with ICT and, thus, creating better cities for all. The definitions of the early 2000's emphasized the streamlining of city operations and optimizing infrastructure through digital services. In addition, the idea of utilising data in decision-making was already

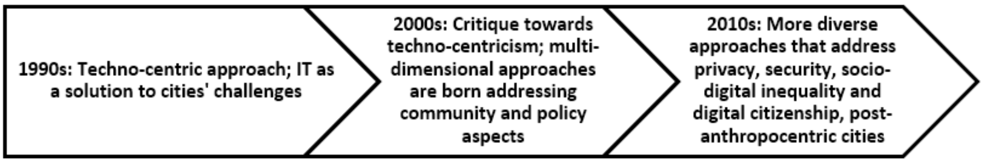

1990s: Techno-centric approach; IT as a solution to cities' challenges
2000s: Critique towards techno-centricism; multi-dimensional approaches are born addressing community and policy aspects
2010s: More diverse approaches that address privacy, security, socio-digital inequality and digital citizenship, post-anthropocentric cities

Figure 4.2: Development of smart city concept.

present in these early definitions [172]. The smart city was at first promoted especially by the private sector, which saw urban ICT systems as an economic opportunity and as a way to work with the public sector [180, 368]. For example, IBM defined the agenda as follows: "Smarter Cities are urban areas that exploit operational data, such as that arising from traffic congestion, power consumption statistics, and public safety events, to optimize the operation of city services. The foundational concepts are instrumented, interconnected, and intelligent." [177].

As the popularity of the agenda increased, also a complete body of work presenting critique towards smart cities' techno-centric approach was born. Many articles suggested smart city agenda can strengthen societal inequalities and lead to unjust cities [84, 191, 272, 341, 410]. Williamson summarized aptly [422], "urban research from geographical and sociological perspectives has sought to critique it [smart city development] in terms of being market-based, technocratic, surveillant, solutionist, militaristic and reproductive of power asymmetries.", see also, e.g. [24, 114, 267]. In other words, the critics argued that smart city development is often realized top-down, without paying attention to city inhabitants' specific needs, perspectives, and local life-words; it follows neoliberal logic; and it undermines ethical questions related to, e.g., free, open public space and privacy. Furthermore, the lack of environmental attributes was repeatedly criticized [280]; or, as Cugurullo puts it, smart city "includes environmental ones as long as they can be monetized" [110].

Due to the increasing critical perspectives, the 2010's definition shows a shift in focus, i.e., the policy and community aspects started to become more common in smart city development and related discussions. According to [129], the redefinition of the term was arguably conducted to distance the concept from the technological determinism surrounding smart cities. One of the central definitions offering a multidimensional perspective on smart cities has been formulated by Nam and Pardo [284]. Nam and Pardo have categorized key conceptual components of smart city into three aspects, including technology (software and hardware infrastructure), human (creativity, diversity, and education), and institutional (governance and policy) [284]. Here, the technology component emphasizes the necessity of well-functioning infrastructure and applications. Without this basis, and therefore, without engagement and cooperation between public institutions, private and educational sectors, and citizens, there is no smart city [284]. The human factors category highlights the value of creativity, learning, and education for the city to become smart. That is, "a smart city is a humane city that has multiple opportunities to exploit its human potential and lead a creative life" [9]. Finally, the institutional dimension emphasizes the fundamental role of a supportive administrative environment (initiatives, structure, and engagement) and governance for the design and implementation of smart city [284]. Therefore, the connection of these factors implies that "a city is smart when investments in human/social capital and IT infrastructure fuel sustainable growth and enhance a quality of life, through participatory governance" [284]. Several researchers have utilized and applied this multidimensional perspective on smart cities. For example, Yigitcanlar et al. [429, 430], have argued that by building on the drivers described by Nam and Pardo [284], i.e. focusing on technology, policy, community, limitations of earlier smart city model(s) could be tackled.

The current smart city literature increasingly addresses aspects relating to privacy, security, socio-digital inequality, and digital citizenship [180, 188, 431]. Further, there exists a strand of research that looks beyond human centeredness and traces the possibility of a smart city model that takes into account non-human beings, i.e., animals and nature, in profound ways [261, 392, 430]. Nevertheless, there seems to be a constant tension between technocentric visions and more holistic visions, and some authors fear that issues that have haunted smart city development already for decades are just carried over to novel data and AI-focused urban visions [110]. Thus, social and environmental themes should always be carefully considered, and plans on digitalization should always be embedded within broader urban policies to avoid one-sided, solutionist, and fragmented approaches [110, 180].

Another view on smart cities is offered by standardization bodies, such as the International Telecommunication Union (ITU) [377] and the International International Organization for Standardization (ISO) [207]. To understand the key components, ITU conducted an analysis of smart cities and sustainable cities definitions [373]. In this analysis, 50 keywords were extracted from 116 definitions found from various sources. Examples of most occurring keywords include quality of life, technology, people, systems, governance and administration, and economy. Therefore, common themes and dimensions were formed from these keywords resembling the six characteristics from Giffinger et al. [153], including Quality of life and lifestyle; Infrastructure and services; ICT, communication, intelligence, information; People, citizens, society; Environment and sustainability; Governance, management and administration; Economy and Finance; and Mobility [373]. This survey helped ITU in identifying key essential terms for the definition of Smart Sustainable City, defined by ITU as "an innovative city that uses information and communication technologies (ICTs) and other means to improve quality of life, efficiency of urban operation and services and competitiveness, while ensuring that it meets the needs of present and future generations with respect to economic, social, environmental, as well as cultural aspects." [373]. ISO 37122[210] provides another view from the perspective of standardization, underlining the role of sustainability: According to ISO, smart city is a city that provides social, economic and environmental sustainability outcomes at increasing pace, and responds to challenges such as climate change, rapid population growth, and political and economic instability. This is achieved by fundamentally improving how the city engages society, by applying collaborative leadership methods, working across disciplines and city systems, and using data information and modern technologies to deliver better services and quality of life to those in the city (residents, businesses, visitors), now and for the foreseeable future, without unfair disadvantage of others or degradation of the natural environment [210].

In theory, these standardization efforts could help in creating a universal understanding of smart city agenda. However, they should be used with caution because standards do not necessarily help in addressing local conditions properly, such as differences in population, economic structures, city management, or social and cultural aspects that can affect smart city development drastically, as mentioned earlier.

4.1.2 . Measuring smart cities

Given the diversity of interpretations, measuring the performance of smart city is challenging [9]. Moreover, cities are very different in their history, culture, economy, and development goals. Therefore, to make the task approachable, quantified measures are suggested that can be tracked over time to give information about stasis and change of a particular phenomenon, i.e. indicators [232]. Kitchin *et al.*[232] distinguishes between single (measuring single phenomenon) and composite (combining several measures) indicators. Also, indicators differ by their role, e.g., descriptive or contextual indicators provide key insights into the phenomenon; diagnostic, performance, and target indicators serve as the means to diagnose a particular issue or assess performance; while predictive and conditional indicators are used to predict and simulate future situations and performances [232]. Here, we first briefly introduce some existing efforts towards measuring smart cities; and then highlight some data-related challenges for such indicators and indices.

A number of standardization and research efforts exist to suggest cities an approach to monitor, analyse, and communicate the performance and progress towards achieving set goals [198, 243], see Table 4.1. For example, the ITU has developed a number of ITU-T Recommendations on assessing different aspects of smart sustainable cities, e.g. [376, 377, 378, 380]. For instance, ITU-T

Y.4903/L.1603 [375, 380] proposes a set of KPIs for assessing cities in achieving smart sustainable goals. This recommendation formed the basis for the development of KPIs for smart sustainable cities by Smart Sustainable Cities (U4SSC). initiative [102]. These KPIs establish criteria to evaluate ICT's contributions in making cities smart and sustainable and provide cities the means to assess the achievements of sustainable development goals. U4SSC indicators form part of a holistic view of a city's performance in economic, environmental, social, and cultural dimensions. Over 100 cities worldwide already implement these KPIs, like Dubai, Valencia, and Moscow [372]. The International Organization for Standardization also puts effort into monitoring and developing sustainable and smart cities. For instance, a number of indicators for sustainable cities and communities were suggested with ISO 37120 [209], which was further complemented with indicators for smart cities with ISO 37122 [210]. There, indicators are broken down by sectors, like economy, education, energy, environment, and climate change. Also, indicators are complemented with metainformation about data sources, interpretation, and calculation methodology. The World Council on City Data is involved in ISO indicators development and provides city certifications based on ISO 37120 indicators implemented [108].

CITYkeys EU Horizon 2020 project focused on the development and validation of key performance indicators and data collection procedures for monitoring and comparison of smart city solutions across European cities [325]. CITYkeys indicators are based on an inventory of 43 existing indicator frameworks and categorised by people, planet, prosperity, governance, and propagation themes [75]. Themes are further broken down into subthemes where 99 project (to assess single projects) and 76 city (to monitor evolution of the city) indicators have been selected and explained in the detail with the mention of expected data sources [75]. What makes CITYkeys project indicators different is that they are impact-oriented. They were also used by the European Telecommunications Standards Institute (ETSI). in their technical specification "Key Performance Indicators for Sustainable Digital Multiservice Cities" [201]. Table 4.2 presents some examples of indicators related to open data and their interpretation within different assessment suggestions.

Having such assessment solutions also allows the creation of indices to enable comparison and monitoring of the city development progress. Indexes can be considered as "quantitative aggregation of many indicators and can provide a simplified, coherent, multidimensional view of a system" [270]. So, these are composite indicators, combining several indicators through weighting or statistics to create a new derived measure [232]. For instance, U4SSC KPIs form the basis for the U4SSC Smart Sustainable City Index that facilitates a comparative ranking of the cities.

Although being useful, the creation and usage of indicators must be done

Activity	Scope
CITYkeys H2020 EU project indicators [75]	Proposes indicators for assessing smart city projects and the corresponding city-level indicators. Indicators are categorised into people, planet, prosperity, governance, and propagation themes, which are further split into subthemes. Altogether, 99 project and 76 city indicators have been presented.
ETSI, Key Performance Indicators for Sustainable Digital Multiservice Cities, ETSI TS 103 463 V 1.1.1 (2017-07) [201]	Proposes indicators based on CITYkeys project [75]. Here, topics include people, planet, prosperity, and governance.
ITU, Overview of key performance indicators in smart sustainable cities, Recommendation ITU-T Y.4900/L.1600 [377]	Gives a general guidance to cities and suggests key performance indicators towards smart sustainable cities, categorised into Information and communication technology, environmental sustainability, productivity, quality of life, equity and social inclusion, physical infrastructure.
ITU, Key performance indicators related to the use of information and communication technology in smart sustainable cities, Recommendation ITU-T Y.4901/L.1601 [376]	Focuses particularly on KPIs related to the use of ICT in smart sustainable cities. Categorised into Information and Communication Technology, environmental sustainability, productivity, quality of life, equity and social inclusion, physical infrastructure.
ITU, Recommendation ITU-T Y.4903/L.1603 [375] and its update Recommendation ITU-T Y.4903 [380]	Proposes KPIs to allow cities to monitor and assess the efforts in achieving sustainable development goals, becoming smarter and more sustainable cities. Indicators are categorised into: economy, environment, society, and culture groups.
ITU, Smart sustainable cities maturity model, Recommendation ITU-T Y.4904 [379]	Proposes maturity model for sustainable smart cities, as well as methods to assess and plan future development strategies. Here, the focus is particularly on assessing the achievement of sustainable development goals towards ICT development of the cities. The proposed model has 5 layers and three dimensions: economic, environmental, and social. KPIs are recommended to be used for assessing maturity levels as well, like published in ITU-T Y.4901 [376], ITU-T Y.4902 [378], and ITU-T Y.4903 [380].
ISO, Sustainable cities and communities — Indicators for city services and quality of life, ISO 37120:2018 [209]	Proposes indicators to assess the performance of city services and quality of life. Indicators are grouped under economy, education, energy, environment and climate change, finance, governance, health, housing, population and social conditions, recreation, safety, solid waste, sport and culture, telecommunication, transportation, urban/local agriculture and food security, urban planning, wastewater, and water.
ISO, Sustainable cities and communities — Indicators for smart cities, ISO 37122:2019 [210]	Proposes indicators to assist cities in assessing the performance of city services and quality of life. Indicators are grouped under the same categories as in ISO 37120:2018 [209].

Table 4.1: Some standardization and research efforts towards measuring smart cities.

Indicator name	Assessment solution	Measurement mechanism	Description
Increase in online government services	CITYkeys project indicator[75]	Likert scale	Indicator analyses the improvement in providing online government services, including open data platforms.
Quality of open data	CITYkeys project [75]	Likert scale	Indicator assesses the ease of use of datasets produced by the project and whether they are kept up-todate.
Accessibility of open data sets	CITYkeys project[75], ETSI[201]	Average stars across all datasets according to the 5 star deployment scheme for Open Data defined by Tim Berners Lee (5stardata.info)	Indicator evaluates ease of use and the openness of city data
Open datasets	CITYkeys project [75]	The number of open government datasets per 100.000 inhabitants	Measures the number of open government datasets
Open Data	ETSI[201]	Number of open government datasets per 100 000 inhabitants	Measures the number of open government datasets
Open data	ITU-T Y.4903 [380]	Total number of open data sets published divided by total number of data sets multiplied by 100	Percentage and number of published inventoried open datasets
Percentage of service contracts providing city services which contain an open data policy	ISO 37122:2019[210]	Total number of service contracts providing city services which contain an open data policy divided by the total number of service contracts in the city, multiplied by 100.	The percentage of service contracts providing city services that have an open data policy
Annual number of online visits to the municipal open data portal per 100 000 population	ISO 37122:2019[210]	Total number of municipal open data portal visits divided by 1/100 000 of the city's total population	Annual number of online visits to the municipal open data portal per 100 000 population

Table 4.2: Examples of open data related KPIs.

with care, since their validity is inbuilt in the process they are created. For instance, indicators themselves describe the characteristics of the system state based on observed or estimated data [270]. This means that the diversity of data sources and measured quality challenges are inbuilt by definition, often making direct comparison unfeasible. Moreover, Kitchin *et al.*[232] emphasise also that data do not exist independently from the ideas, interests, technologies, practices, and systems involved. Therefore, they should be used and interpreted with caution. All these imply that assessment frameworks should provide a clear description, rationale, interpretation, benchmarking, and methodology for indicator calculation, as well as potential sources of possible data to use and links to other normative documents [102]. So that the one using the framework is equipped with all the information regarding data. Also, indicators can show that a problem exists, but they do not show its cause or tell what to do [232]. Therefore, they could be useful if monitored continuously, to see the progress if certain measures are taken. This also raises the question of whether a city index to rank the cities is needed [74]. Given the fact that cities are very different from each other and have diverse histories, economics, and development goals, their ranking can be misleading and provide weak support for cities themselves in their development. Moreover, "indicators and measurements should not become a goal in themselves but support the fulfillment of individual cities' needs" [198]. From this perspective, indicators supporting continuous monitoring of important phenomena in the city could be valued more. Also, indicator visualization is important, since this may affect perception and interpretation [232, 286].

Indexes should also be used with caution. For instance, indices usually have a certain focus, which determines which indicators are included in it [74]. It is recommended to develop a solid theoretical framework to serve as the basis for the selection and combination of indicators into a meaningful composite indicator [286]. Therefore, developers of the index should understand and communicate the purpose and limitations of the index, as well as how different indicators relate, so that index interpretation is solid [270].

In addition, indexes also rely on a number of data processing techniques, like aggregation, normalization, and weighting [364, 286]. Proper theoretical grounds should be followed, otherwise " 'incompatible' or 'naive' choices (i.e., without knowing the actual consequences) in the steps of weighting and aggregation may result in a 'meaningless' synthetic measure " [156]. Moreover, it is recommended to test the aggregate measures for their robustness as a whole, to test how sensitive is the index to changes in the steps followed to construct it, traditional techniques include uncertainty and sensitivity analysis [156, 286]. These imply that data, overall methodology, predetermined boundaries of the system, and comparability of results across the systems should be transparent and clearly communicated so that one is able to assess the performance and suitability of index for particular task [270]. Finally, aggregate indices may hide some information, e.g., it could be challenging to identify if few indicators have extreme values when the index aggregates hundreds of these into one number [270]. In such situation, it could be better to provide the indicators as frameworks and use, maybe, visual tools to present these.

Indexes require careful governance as well, as with the time, data behind indicators can change, therefore direct comparison with previous versions becomes unfeasible. The ability to compare various indicators and assessment frameworks provides means to ensure that the proper one is selected. Huovila *et al.*[198] provide a comparative analysis of standardized indicators for smart sustainable cities, where seven sets of city indicators published by international standardization bodies are inspected in terms of their conceptual urban focus, city sectors, and types of indicators.

Acknowledging the limitations and challenges, indicators are still useful and provide the means to track the progress of certain phenomenon [105, 335]. The key message here is to enable as transparent and documented process as possible, ensuring that users of the indicators have a proper understanding and are able to make an informed judgment if the indicator is suitable for the task at hand.

4.1.3 . Smart city architectures and platforms

Smart cities are very complex structures involving various stakeholders, technologies, and physical constraints; therefore, it is difficult to provide a unified reference architecture and a platform, since the development could be guided by its own requirements [252, 350]. In this section, we'll cover some existing efforts towards smart city architectures and platforms and summarize them into a general architecture from the smart city data point of view.

ITU defines architecture in general as "a definition of the structure, relationships, views, assumptions, and rationale of a system" [374]. There are many smart city architectures and their implementations presented by the research community, varying in their goals and details. Generally, smart city reference architectures should be technology-neutral and provide a clear set of capabilities and stages to be implemented in order to provide smart city services [145]. Moreover, such architectures aim to fulfill a certain set of requirements of the domain. Table 4.3 summarizes requirements for smart city architectures and platforms found in related work. As can be seen, in general, such requirements cover data and system management functionality, as well as non-functional requirements related to privacy, security, and system lifecycle management.

A number of architectural proposals exist with varying levels of detail. Some researchers provide quite a general perspective. For instance, Zygiaris [443]

Functional requirements	Non-functional requirements
Summary from [350, 374, 163, 264, 69, 252, 142, 203, 398, 317]	
• Handling Big Data characteristics, namely Volume, Velocity, Variety, Veracity, and Value • Definition of a City Model, data models, and APIs • Data management • Data storage management • Data Processing & Analysis • External Data Access • Applications Runtime management • Wireless Sensor Network Management • Service Management, SLA • Software Engineering Tools, APIs • IoT device/resource discovery and management • IoT Data Marketplace • License management • Incorporation of Feedback and Monitoring	• Interoperability • Decoupled & distributed components • Openness • Legacy Compatibility & heterogeneous landscape • Resilience to failure & Robustness • Performance • Scalability • Security • Privacy • Context Awareness • Adaptation • Extensibility • Configurability

Table 4.3: Summary of requirements for smart city architecture and platform from related work.

suggests seven layers, going from the layer covering essentials of the city (districts, inbuilt infrastructure, etc), to level aiming and promoting green and sustainable actions (like green transport practices, and planning), to technology and application covering layers (interconnection, instrumentation, open integration, application layers), and, finally, to innovation layer, focusing on innovation ecosystem vital for the prosperity of the cities and their inhabitants. Zheng *et al.* [437, 438] summarize the urban computing system framework, which is comprised of four general layers: urban sensing and data acquisition, urban data management, urban data analytics, and service providing. In contrast to other proposals, Zheng *et al.*[437] are more interested in methodological aspects, like processing geo-spatial data at each layer (e.g., trajectory compression and map-matching in the urban data management layer).

Others focus more on the system development angle. For instance, Habibzadeh *et al.* [163] abstracts smart city architecture as five generic planes: application plane, sensing plane, communication plane, data plane, and security plane. There, each plane comprises a number of technologies, methods, and challenges. Santana *et al.*[350] provide their reference architecture for the development of software platforms for smart cities based on analysis of 23 related projects. Compared to others, their architecture is more technology-driven and is based on the cloud and networking layer, with Internet of Things (IoT) and Service middleware, user management, and social network gateway on top of that. The Big Data management component is responsible for all data aspects. In addition, the need for the toolkit, security support are presented in the architecture. The authors also emphasize that all components of the platform must support scalability, security, privacy, and interoperability. Santos et al. [351] focus on sensing platform for smart cities. Their approach is to follow the data flow: sensing, data collection, and data storage, processing, sharing, and hosting urban services. They integrate sensor data from mobile crowdsensing, environmental, and public transport vehicle sensing for analysis, data sharing, and smart city applications development. There, the importance of a unified spatio-temporal data model and the use of standard IoT data access methods are emphasized. Villanueva and al.[411] propose Civitas platform to be seen as the core of smart city IT infrastructure able to orchestrate different entities (like citizens, public institutions) connected to it via Civitas plugs. Middleware also relies on core nodes that are servers hosting a variety of services. Authors emphasize the integration of intelligence, like common sense reasoning. When compared to others, this proposal is more broker-like. Bibri [69] provides an analytical framework for data-centric IoT applications for smart sustainable cities. Their proposal provides a pipeline focused on IoT, Big Data, Cloud, and Fog programming paradigms. Its main components include Urban systems and domains that should function and be managed by IoT and its underlying big data analytics; Urban big data sources, storage facilities, and data categories component is responsible for data collection, storage, and management; Cloud computing or fog/edge computing and Hadoop MapReduce architecture infrastructure for big data processing and management to for knowledge discovery/data mining; Big data applications covers smart applications for diverse urban domains [69]. CUTLER EU project proposes a data hub conceptual architecture to support data management and analysis for decision making in municipalities [398]. In comparison to other proposals, they provide quite a general data-centric conceptual solution, which is then illustrated with concrete implementation for five pilot cases. Their main blocks in architecture are: data collection, representing data acquisition functionality (like data sources, data crawlers, data preprocessing); data integration platform supporting data ingestion, data storage, and access APIs to other components that will further manage and/or process the data; data analytics to support business logic of the smart city services; data governance to manage the data and data lifecycle; business model DevOps to bridge the gap between the big data technology and the business model of policy developments; and services & visualization responsible for smart city services and data visualization [398]. Similarly, Pereira et al. [317] suggest a platform for integrating heterogeneous data and aiding the development of smart city applications. In comparison to other proposals, their solution emphasizes a semantic-based data model. For example, in their proposal, information is grouped into layers that represent geographic or some particular domain information, like School or Public safety. The information from different layers could be linked together to retrieve new information, e.g. information about safety close to schools. Architecture-wise, it is a distributed system consisting of SGeol middleware and middleware infrastructure, which includes components for managing users and data access security policies; managing data, its messaging, integration, and context; discovery of physical devices and their integration to the platform; real-time and batch analysis. The solution also provides RESTful APIs for external data access and SGeoL Dashboard service offering edit, query, and visualization capabilities.

SynchroniCity EU project (that included also partners with leading roles in standardization bodies) aimed to establish a reference architecture for the IoT-enabled city marketplace, ensuring interoperability and developing interfaces and data models for different verticals [264]. To achieve that, SynchroniCity project analyzed available models and approaches for smart cities and summarised them with an architecture framework collecting the most common capabilities and technologies [264]. Their reference architecture consists of different logical modules, including Context Data Management to manage the context information coming from various data sources; IoT Management module responsible for interaction with the devices using different standards or protocols to make them compatible with the framework; Data Storage Management responsible for data storage and access; IoT Data Marketplace facilitates business interactions between data suppliers and consumers by enabling digital data exchange; Security, Privacy and Governance module handles security aspects related to data, IoT infrastructure and the platform services; Monitoring and Platform management services module guards platform configuration management and service activities monitoring; Southbound interfaces to connect the architecture to various data sources and IoT devices; Northbound interfaces provide platform functionalities to be used by the final smart city end-user applications [264].

Standardization bodies are also interested in providing architectural solutions enabling smart cities and they have close views, as the SynchroniCity project. For instance, ITU provides different angles on smart sustainable city reference architecture. Their ICT architecture from a communication view, emphasizing the physical perspective, relies on the top of the city's physical infrastructure. This architecture consists of sensing, network, data and support, application, and operation, administration, maintenance and provisioning, and security layers. Architecture also demonstrates communication and exchange of information between the layers [374]. ETSI puts context management and interoperability at the core of their platform [203]. They suggest a smart city platform that is based on the NGSI-LD ecosystem [202, 204]. The main logical functions of their framework are as follows. Data ingestion & integration to collect data from different systems; NGSI-LD Context Broker applying NGSI-LD API [204] for data interoperability; Semantics for construction and use of semantic data and technologies; Analytics & Artificial Intelligence to support analysis/prediction services for smart cities; Monitoring & management responsible for system operation monitoring and management; Security & Access Control is responsible for authentication for smart city platform users and applications, access control policy management, and access control token management functions [203]. Their architecture also considers data spaces, through Data space connector smart cities can connect to other data spaces and share data across other relevant systems[203]. Similarly, with a focus on context management, FIWARE suggests reference architecture for smart cities. Their architecture is technology-oriented, where Orion Context Broker is its core component. FIWARE provides data models, interfaces, and ready-made components for e.g., IoT, processing, analysis, and visualization of data [142]. For instance, the FIWARE platform was used to provide the main components for underlying middleware infrastructure for SGeol middleware [317].

In Figure 4.3 we summarise the essential functional blocks required for a smart city data platform. Logically, we divide the architecture into data sources, platform, and applications. Data sources represent possible data that can be used for the development of smart services. The platform incorporates a tra-

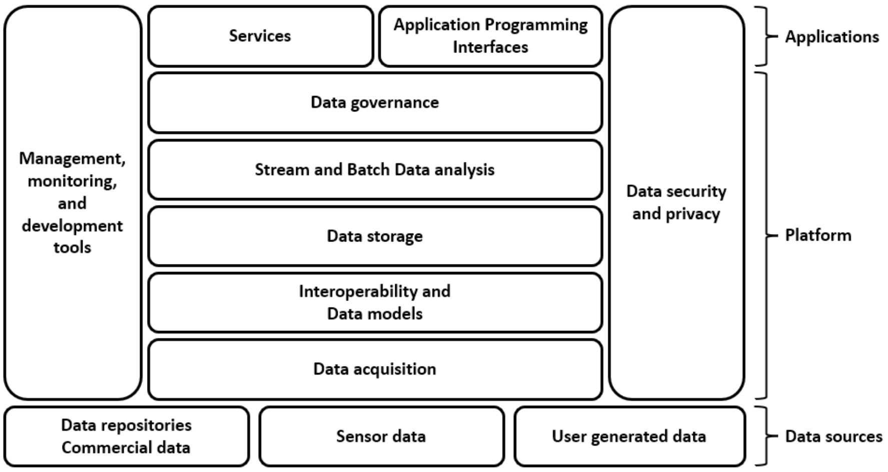

Services
Application Programming Interfaces
Data governance
Stream and Batch Data analysis
Data storage
Interoperability and Data models
Data acquisition
Management, monitoring, and development tools
Data security and privacy
Applications
Platform
Data sources
Data repositories Commercial data
Sensor data
User generated data

Figure 4.3: Smart city data system architecture, the summary from related work.

ditional data management pipeline. The important aspect here is interoperability and data models, as pointed out by some related work [142, 264]. Traditional blocks also include data storage and analysis. The data governance functional block ensures the overall usability of data assets in the platform. Data security and privacy are the backbone of the platform. Finally, management and development tools are needed to ensure that the platform is operational. On top of that, the development of application programming interfaces would facilitate accessing data/analysis results or performing certain actions. The services block represents numerous services that could be developed on top, like smart transportation services.

Concrete implementations of such functional blocks could vary greatly, from more centralized cloud-based solutions to more distributed ones, like edge-based [228, 319]. Therefore, methods and tools would be selected accordingly. For example, architectures and platforms are proposed to support the development, deployment, and management of IoT systems across a number of devices with varying resources, e.g., Osmotic Computing Platform [412]. An in-depth review of methods and technologies for concrete implementations of smart city data architectures, as well as deployment and management frameworks, is out of the scope of this research. For such studies, refer to [163, 319, 350]. Instead, we focus on data challenges from a more conceptual standpoint, leaving their concrete implementation and selection of methods and tools to the developers.

4.2 . Data challenges in the context of smart cities

The development of IoT and communication technologies opened up numerous opportunities to assess a variety of phenomena in cities, like traffic, pollution, and economic wealth. City data is diverse in nature and has a variety of formats, availability, volume, spatiotemporal dependencies, and sensitivity concerns, to name a few. All this data should be processed and analyzed to derive comprehensive insights. Therefore, solutions are needed to work with such diverse data in a robust, efficient, secure, and ethical manner. This section reviews the main issues and approaches developed in smart cities context in *(i)* data availability and quality, *(ii)* data heterogeneity and integration, *(iii)* data management *(iv)* data analysis, *(v)* ethics, *(vi)* data privacy, and *(vii)* data security.

4.2.1 . Data availability and quality

Different taxonomies were applied to classify urban data. For instance, Zheng et al.[438] suggest a division of urban data by the nature of phenomena they present, like geographical, traffic, mobile phone signals, commuting, environment monitoring, social network, economy, energy, and health care data. Another suggested taxonomy is based on data structures (point- and network-based types of data) and spatiotemporal properties (spatiotemporal static, spatial static but temporal dynamic, and spatiotemporal dynamic) [437]. Also, available urban data can be divided into five pools, including firewall (within the legacy systems of public agencies), open data, social, sensors/IoT, and commercial data [144]. Finally, urban data was also divided based on including personal information, like non-personal data, aggregate data, de-identified data, and personal information [242]. In this subsection, we will highlight the urban data availability aspect, categorizing our exploration into open data, citizen-contributed data, and commercial data solutions. Also, we will discuss corresponding data quality considerations.

Open data. Data is the key enabler for the vision and realization of smart cities. According to a European strategy for data, Big Data is considered as one of the key enablers to maximise the growth potential for the European digital economy and society [103]. Therefore, a significant effort is made to promote data suppliers and owners, even municipalities and governments to open their data for both research and business. To gain the benefits, an adaptation of municipal vision and governance strategies could be required to coordinate, enable, and support various forms of data-sharing initiatives [233]. Open data is the data that anyone can access, use, and share; it is available in machine-readable format, as well as licensed to permit data use in any way [104]. Actually, governments and municipalities play crucial role in the management of cities' data assets to be able to use the data-driven tools to address the city challenges [57]. Therefore, there is also a strong recent trend to release much of public agencies data as open data [144], so-called

Open Government Data, defined as information collected, produced, or paid for by the public bodies and licensed for free re-use for any purpose [104]. A number of open-source and commercial data portal platforms exist, providing abilities to publish data, enabling data access and visualizations like CKAN[1], DKAN[2], Socrata[3], Opendatasoft[4], PublishMyData[5]. Their availability, as well as the strong demand to share urban data, has resulted in a number of urban data platforms, containing both open and restricted in-use data. Barns [57] classifies these into data repositories - open data portals with the main goal to provide data sharing capabilities; data showcases that aim to visualize data, but the data itself is not always available or machine-readable; city scores - visualization of city performance in regard to a certain set of indicators; and data marketplaces enabling data access and reuse with performance monitoring. Examples of data repositories include, e.g., New York City open data portal [113], which enables data access within a number of categories. Among the full information about the dataset, it is also possible to see the data snapshots and visualize the data in external services. Another example of data repositories is Moscow City Government open data portal[6], providing access to the data classified into thematic topics, like healthcare, education, and culture. Datasets are equipped with basic information, like, among others, dates, formats, links to the source, and contact information of persons responsible. Well-known city dashboards include Dublin Dashboard[7], which provides rich visualization opportunities as well as possibilities to get the data available. London Datastore [258] also provides reach opportunities to visually explore the data, as well as get access to it. However, when compared to other city dashboards, the London Datastore provides data-driven analytics based on their alignment to strategic planning and governance challenges for City Hall [57]. Table 4.4 gives brief summary of selected available datasets. For deeper insights, an interested reader could refer to Ma et al. [262], who survey available city datasets.

There are a number of initiatives in EU advancing data sharing. For instance, open data portal[8] provides access to data published by EU institutions and bodies. In addition, portal provides opportunities for data visualizations and work with linked data. Also, European Data Portal harvests the metadata of public sector information available on public data portals across European

[1] https://ckan.org/

[2] https://getdkan.org/

[3] https://www.tylertech.com/products/socrata

[4] https://www.opendatasoft.com/

[5] http://www.swirrl.com/

[6] https://data.mos.ru/

[7] https://www.dublindashboard.ie/pages/index

[8] https://data.europa.eu/euodp/en/home

Dataset	Summary
City Pulse Aarhus City [11]	Dataset provides information related to traffic observations, weather situations, pollution, and cultural events from the city of Aarhus, Denmark. The dataset has been used, e.g., to forecast traffic situations, study privacy concerns, measure air pollution, and develop transfer learning algorithms [356, 19, 192].
Amsterdam [20]	The dataset measures traffic, accidents, crime statistics, economic activity, and pollution. It has been used, e.g., to estimate the effect of parking prices, forecast traffic flows, fast charging planning for vehicles, and contextualisation for sustainable development [309, 62, 182, 215].
Chicago Datasets [99]	Datasets include traffic congestion estimates, traffic counts, accident and emergency dispatches, energy usage, air and water pollution, and data related to economic activity. The dataset has been used for, e.g., forecasting daily crime, traffic prediction, studying residential energy efficiency, and crime analysis surveys [439, 2, 352].
London [258]	Greater London Authority provides a wide range of data related to traffic counts, street crime, energy usage, data related to borough profiles, topsoil chemical data, wealth gap, and birth trends [257, 262]. The dataset has been used to, e.g., analyse crime patterns, forecast energy usage, and borough-level COVID-19 forecasting [307, 332, 126].
New York [113]	The portal provides data related to vehicle collisions, crime data, energy and water data, air quality, water quality complaints, school districts, enrollment statistics, and others [113]. The data has been used in different studies to asses the needs after Hurricane Sandy, electricity estimation, crime prevention, study air pollution trends, and predicting burglaries [137, 440, 221, 369].
AirNow [4]	AirNow platform provides air quality data about local areas in the United States, Canada, and Mexico from more than 500 locations [4]. The data has been used to study and forecast wildfire pollution, bias correction in air quality forecasting models, ozone forecasting, and the effect of ozone on children's health [342, 265, 173, 13].
Tokyo Open Data [391]	Tokyo Metropolitan Government has developed an open data portal to provide insights to different city segments. The platform provides case studies, data related to bus stations, disaster prevention maps, and data related to the environment (e.g., air pollution, landfill, sewerage, etc.). The data portal has been used, e.g., to organize a hackathon to address administrative issues, analyze social trends related to COVID-19, investigate the crime harm index, and study issues related to the lack of educational data [400, 387, 189, 303].

Table 4.4: Summary of selected datasets.

countries[9]. Other data sharing activities include INSPIRE Geoportal[10] that collects data provided by EU Member States and several EFTA countries under the PROCLAI, which focuses on creating an infrastructure for sharing environmental spatial information. Yet another known initiative is Copernicus, the Earth observation programme coordinated and managed by the European Commission and is implemented with the Member States, the European Space Agency, the European Organisation for the Exploitation of Meteorological Satellites, the European Centre for Medium-Range Weather Forecasts, EU Agencies and Mercator Océan[11]. Copernicus provides a number of services categorised under atmosphere, marine, land, climate change, security, and emergency themes, as well as access to satellites and in situ sensor data.

Acknowledging the power of such dashboards and portals, they require considerable effort to remain useful and provide utility for communities, municipalities/governments, and businesses. First, their purpose and interpretation should be as clear as possible, since the data itself, as well as data processing and analysis steps are known to be technology and methodology dependent, limited in time and location, and could be biased in interpretation [231, 232]. Second, such data platforms require active maintenance and support to ensure that they contain up-to-date information of the required quality. Support is also needed for both data providers and data consumers. For instance, proper effort is required to share the data. Data provider must ensure the content quality (completeness, cleanness, accuracy), timeliness and consistency support, data representation model (use of standardised solutions, proper formats, linked data), supply of proper metadata, as well as, addressing the legal aspects, i.e. to provide a license to use the data [104]. After data is published, it should be properly maintained, i.e. checking data access and assessing and updating data itself and its metadata, as data lineage and metadata allow users to assess the trustworthiness and data quality [232].

Legal issues regarding publishing and use of the data require careful treatment. For example, data ownership, legal grounds, and terms of use are often unclear for particular data sources within data repositories. Many data repositories have statements and references to legal documents in terms and conditions on what kind of data is stored and how to use it, e.g. Moscow City Government open data portal. However, e.g. including licence information in data source description itself provides better transparency and eliminates confusion, check the London Datastore for example.

Citizen-contributed data. The premise of citizen-contributed data is to facilitate and collect input for decision-making at large. Different approaches exist to harness citizens' data [239], including

[9] https://www.europeandataportal.eu/

[10] https://inspire-geoportal.ec.europa.eu/

[11] https://www.copernicus.eu/en

- *crowd markets*: to enable aggregation of online individuals as collaborative input;
- *social media mining*: to retrieve publicly expressed opinions and content;
- *urban and in-situ sensing platforms*: to unobtrusively collect data from citizens' daily dwellings.

Crowd Markets. Amazon's Mechanical Turk [17] and Figure-Eight [132] (previously Crowdflower) are today's largest platforms for aggregating online individuals' time to complete tasks that are computationally intensive but relatively trivial to a human. These platforms are purposefully generic, and a variety of tasks can be created. These tasks range from answering to surveys, writing reviews, annotating images, transcribing audio, and others, i.e., tasks that are challenging to use computers for automation due to a high risk of error. The main challenge of crowd markets is to sustain the crowd size and quality. Literature shows that higher-paid tasks can attract workers at a higher rate. Emphasis on the importance of the work has a statistically significant and consistent positive effect on the quality of the work [338]. A practical example of leveraging crowd markets is Zensors [433], which enables sensing from any visual observable property. Zensors streams images where the crowd processes and labels according to a well-defined set of instructions, enabling near-instant counting and another high-level sensing. Once sufficient human-based input is available, machine learning is applied to fully automate the process once the accuracy of the algorithms is high (>90%). This approach is also used by Google Crowdsource initiative [155], where gamification and recognition as badges are used to sustain and train machine-learning classification algorithms.

Social Media Mining. Online social media mining on a large scale allows us to consider users' posting of opinions and content in online social media to gain insight into unfolding events [338]. The widespread availability of smartphones and high-speed internet has enabled a range of systems that collect a variety of different types of user contributions. For example, it is now possible to collect videos and photos on the field, e.g., YouTube, Instagram, Twitter, and Facebook. These platforms allow user-driven tagging with relevant keywords. The primary use of this media is for the platform, but researchers have found such user-generated content as sensor data, originating from end-users. Providing a system that allows users to easily contextualize and tag high-level data results in a valuable repository of knowledge. For example, Wheelmap[12] allows users to tag, search for wheelchair accessible places using one's smartphone and browser. Others share where they are [407] or

[12] https://www.wheelmap.org

whether that place is recommended [238], or reported the destruction aftermath of an earthquake [413]. Researchers keep exploring ways to use devices' sensors usage, as Citizen Science [314]. Citizen Science can be interpreted as individuals becoming active participants and stakeholders of data. Large-scale efforts, such as Wikipedia, OpenStreet Maps, allow users to publicly augment and annotate online information as text or geo-fenced markers. This wealth of everyday information about and around us creates numerous possibilities for new applications and research in general. Social media-enabled applications are primarily driven by smartphones for in-situ context and are often deployed on application stores for ease of installation and updating the platform.

Urban and in-situ sensing platforms. Urban and in-situ systems pervasively collect data from citizens without the need to set up or install an app on someone's smartphone. These platforms often deploy sensors throughout a city. These can be invisible to the citizens, e.g., underground traffic sensors, weather monitoring stations on top of a building, or can be an integral part of the city, e.g., interactive public displays. A number of studies have investigated the use of public interactive displays for the purpose of data collection [21, 77, 193]. Opinionizer [77] is designed and placed in social gatherings (parties) to encourage socialization and interaction. Participants would add comments to a publicly visible and shared display. Due to fear of "social embarrassment," the authors suggest public interactions to be purposeful.

The environment, both on and around the display, also affects the use and data collected. The environment produces strong physical and social affordances, which for facilitating the public, they need to expose their purpose towards the social activity rapidly and to be able to encourage seamlessly and comfortably a citizen from being an onlooker becoming a participant. Text-Tales [21] explored providing story authorship and civic discourse by installing a large, city-scale, interactive public installation that would show a grid of text. A discussion on a certain photograph would start with SMSs sent by the citizens, displayed on a stream of comments.

Beyond a public display, citizens can also be involved in larger efforts to affect society at large. Projects such as vTaiwan[13], an online-offline consultation process that brings together government ministries, elected representatives, scholars, experts, business leaders, civil society organizations, and citizens. The platform allows lawmakers to implement decisions with a greater degree of legitimacy. It combines a website, meetings, hackathons, and consultation processes. For example, vTaiwan was crucial in the debate of Uber operations in Taiwan[14]. In a similar approach, Decidim[15] is a digital platform for cit-

[13] https://info.vtaiwan.tw/

[14] https://vtaiwan.tw/topic/uberx

[15] https://www.decidim.org

izen participation, helping citizens, organizations, and public institutions self-organize democratically at scale. It provides a political network, citizen-driven initiatives and consultations and raises participatory budgets, thus allowing a democratic and flexible system where everyone can voice their opinion.

Overall, citizen-contributed data is a very valuable source of information, and in some cases, it is the only way to understand the phenomenon of interest. However, such data collection initiatives and subsequent data analysis should be planned well and performed with care. For instance, if citizens are asked to do a measurement, they should be instructed on how to do it to get reliable value [78]. Some measurements may also require a calibration of the device [321]. In addition, one should have a strategy to deal with data gaps due to behavioral patterns of people doing measurements [333]. As in each study, one should ensure that a sample of users, contributing the data to the system, represents the population as fully as possible, and no bias is introduced into the data collection strategy. Finally, privacy issues from such data collection initiatives should be discovered and treated appropriately.

Commercial data and private-public partnership. A number of commercial organizations deploy infrastructures and utilize available urban data to provide and improve their services. Sharing these data with municipalities has been a question of debate for a long time [401]. However, challenges with data enabled various forms of commercial involvement, like data markets and hubs. Such organizations facilitate connections between data providers and data consumers, especially if the data cannot be openly shared. One example of such a solution is Platform of Trust[16], Finland, that enables data movement between systems and organizations, taking care of trustworthiness and data harmonization issues. They also involve the community so that interested people can participate in creating harmonization models that are then published as open-source code.

Also, possibilities are explored for public and private organizations data exchange, e.g., City Data Exchange (CDE) project created a marketplace for public and private organisations data exchange [299]. This project was a collaborative effort of the Municipality of Copenhagen, the Capital Region of Denmark, and Hitachi. CDE service provided collaboration between different partners on supply and demand of data and a platform for selling and purchasing the data for both public and private organizations. Based on the project, a number of challenges were identified, e.g., immature market as even though some companies buy the data for their services, generally many are not yet ready to include data sharing into their core business or strategy; lack of use cases seems to affect the reluctance to invest resources in selling/buying the data; fragmented landscape; reluctance to share data on an open data portal, e.g. due to ethics or competitors' advantage reasons; lack of skills and

[16] https://www.platformoftrust.net/

competences to work with data [299].

Development of such joint efforts requires trustworthy data stewardship. That is, “trustworthiness is the virtue of reliably meeting one’s commitments, while trust is the belief of another that the trustee is trustworthy” [302]. Several models are suggested to collaborate in data use and share [205]. For example, data collaboratives[17] represent a form of partnership where a number of parties, like governments, companies, and others, collaborate to exchange and integrate the data to help to solve societal problems or create a public value [235]. Therefore, through such cross-sector and public-private collaboration initiatives it is possible to achieve much wider goals that are difficult to perform by the parties by themselves only. One noteworthy example of data collaboratives in smart city context is ’9292’[18] that is public-private collaborative, gathering and sharing public transportation data in Netherlands. Obviously, data collaboratives possess all the challenges that data integration initiatives have, since the data comes from diverse providers, in different format and structures. However, as Klievink et al. [235] emphasize, data collaboratives are collaboration and innovation phenomenon rather than data phenomenon. Therefore, organisational, incentivisation, and governance challenges should be considered as well. From this perspective, a number of additional challenges arise regarding vulnerabilities in opening the data, its possible misuse, and overall trust within partnership. Coordination problems also include matching potential data providers and data users, maintaining data control and its unforeseen uses when shared, matching a problem with the data attributes, ensuring the shared data is useful and usable by the user, aligning incentives of providers to share proprietary data with the goals of the users [386]. Moreover, data collaboratives are not isolated constructs, therefore partners’ incentives, goals and collaboration overall depend on context, like institutional and governance frameworks, government interests, transparency/inclusiveness culture, and means by which collaboration is legitimised [235]. Therefore, to have a successful collaborative, it could be helpful to organise the overall collaboration process and context in such a way that perceived vulnerabilities are dealt with [235].

Another initiative is data trust. The interest in data trusts was coined in 2017 where this model was proposed as a “set of relationships underpinned by a repeatable framework, compliant with parties’ obligations, to share data in a fair, safe and equitable way” [171]. Open Data Institute defines data trust as “a legal structure that provides independent stewardship of data” [176]. There are a number of interpretations of data trusts, e.g. it is assumed that data trust could be simply an arrangement of governance or a legal agreement or such practices could be aggregated into architecture [302]. Hardinges places

[17] http://datacollaboratives.org

[18] https://9292.nl/en

different interpretations and uses of data trust term into the following categories, including repeatable framework of terms and mechanisms; a mutual organisation formed to manage data on behalf of its members; a legal structure; a store of data with restricted access; and public oversight of data access [175]. For instance, Sidewalk Labs proposes the establishment of an Urban Data Trust (that could evolve into a public-sector agency over time) serving as an independent digital governing entity for their Sidewalk Toronto project, ensuring that responsible data handling is in place for digital innovation activities (Responsible Data Use) [242]. In addition to privacy laws, Sidewalk Labs suggests that all innovations aiming to collect/use urban data must go through Responsible Data Usage Assessment conducted by Urban Data Trust. This way, Sidewalk Labs aims to achieve the proper privacy and security practices, provide and use consistent and transparent guidelines for responsible use of data, and make urban data a public asset [242]. These goals align with O'Hara's emphasis on the purpose of data trust, which is "to define trustworthy and ethical data stewardship, and disseminate best practice" [302].

Generally, successful engagement in any form of data-sharing partnership could require adaptation of urban governance visions and strategies [233], as well as transformation of parties' institutional cultures and processes [121]. Though, a certain level of data quality could be expected from commercial or private-public partnership data, since such data often is an asset for the commercial success of organizations. However, the technological and methodological biases should not be excluded, since the data could be generated for a particular purpose, but shared for potential other ones [231, 232]. Moreover, partnerships could suggest proper formalization of the responsibilities in data sharing (e.g., data representation models and metadata availability), usage (e.g., who, how, for what purpose), and maintenance processes between collaborating parties, making sharing and usage of the data smoother.

4.2.2 . Data heterogeneity and integration

During the last few years, a large amount of heterogeneous data has been available from various applications and tools. This is also true in the smart cities context, where rapid adoption of intelligent applications has created new, different, and numerous data collections. These new sources have given new opportunities but also emerging challenges. An effective data analysis in the smart cities context has to consider the increasing amount of data coming from connected devices, multiple software (developed by public and/or private institutions), and historical archives. However, since the systems producing and collecting data are heterogeneous, they provide data in multiple formats that must be integrated to be combined for running an effective analysis. The siloed and often incompatible nature of these sources has also made the interpretation and use of data more challenging [328]. We will explore the different strategies that, according to the literature, can be applied for integrating data focusing on smart cities, summarized in Table 4.5:

- Model data integration
- Semantic data integration
- Structural data integration
- Software-delegating data integration

Model data integration. This approach for data integration has been developed in the previous decades starting from proposals focused on the integration of classical data models (such as Relational, XML, and Object-Oriented) [167, 271], and continuing with suggestions more focused on recent data formats (such as streams, NoSQL databases) [47, 245]. According to this methodology, all data, coming from different sources is collected in a central repository where an abstract model, grouping all the characteristics of the diverse sources, supports all the operations [65]. A major benefit of this methodology is the fact that data collected and integrated (in theory) contains no redundancy, can be accessed uniformly, and can be trusted thanks to its integrity. Unfortunately, the definition of such a model is difficult since integrating concepts coming from different data models is not always easy. For example, it could be quite challenging to integrate into the same model two dissimilar concepts, such as a link from a graph data model and a column from a columnar data model. Moreover, the characteristics of Big Data make the maintenance of such a unified model tricky since the data model must be updated each time a new data source with a different data model is defined and needs to be integrated.

In the context of smart cities, the work of Ballari et al. [54] presents one of the first approaches in this direction. The authors focus on integrating sensor data and highlight the difficulties in finding a global scalable solution. Even though they introduce a global model (providing dynamic interoperability and considering the concepts of proximity, adjacency, and containment in different dynamic contexts), they still cannot manage to introduce a global schema that can be used to store data in a scalable manner. The CitySDK project [318] goes in the same direction, defining a global data model for integrating data concerning tourist information. Their global model designs structures for points of interest, events, itineraries, and categories/tags. The approach bases the data collection on a set of adapters that transfer the information from the heterogeneous sources (mainly CVS, JSON, and XML files) to the global data model (implemented in document format and stored in MongoDB) using a REST API. This approach tries to solve the problem of the flexibility of the central data model by requiring the definition of a new adapter each time a specific data source is added to the system.

Model	Semantic	Structural	Software-delegating
Data Integration	Data Integration	Data Integration	Data Integration
All data belongs to a unified schema in a target meta-model	A general domain ontology represents all the concepts	Data integration occurs at the physical storage level	Off-the-shelf software is used for integration
+ Unified vision of data	+ Modularity and scalability	+ Transparent to the high-level analysis	+ Ready-made solutions
+ Allows to identify and possibly eliminate data redundancy	+ Easy and transparent integration of new data sources	+ Unified and efficient data access patterns	+ Modular solutions: new developments easily extend models
+ Algorithms defined in a general way on the global schema	+ Reasoning on objects and their relationships	+ Operations at data-fragment level that can scale-up easily	+ New analysis can be included with new components
- Users must have a high capacity of abstraction	- Domain expert knowledge is required	- Security and privacy are fully delegated	- Data access depends on the platform and its capabilities
- Usually, standard query languages are not available	- Already-available ontologies do not always fit the target scenario	- Access from external software and platforms is not easy	- Updates from vendors can affect the global design
- A new data source can impact the general model	- Poor support for stream analysis	- Data must show a uniform storage format and granularity	- Strong dependency on platform capabilities
Examples: [54, 79, 236, 271]	Examples: [353, 68, 71, 106, 112, 150, 157, 327]	Examples: [107, 126, 320, 328, 329, 334, 337]	Examples: [118, 336, 362, 328]

Table 4.5: Data Integration strategies in smart cities, with their benefits (+) and challenges (-).

More recent approaches have managed to establish architectures based on the meta models provided by new technologies. This is the case of the data hub-like architecture, proposed by Koh *et al.* [236]. This approach integrates the technologies of stream processing, like Apache Kafka [35] with the support of Apache Spark [39] (also used for batch processing); the knowledge graph-structured base of Virtuoso for semantics, and the storage of Apache HBase [34] for quick real-time retrieval. Finally, they use Vert.x [409] a Java framework to provide scalability through its natively asynchronous task processing and abstraction of microservices. The design is still quite new and would have to be tested to evaluate its performance.

Cacho *et al.*[79] proposed viewing a smart city as a system-of-systems (SoS) in order to help develop a framework upon which governments can benefit from the integration of public and private systems for planning, administrative, and operative purposes. They also identify a few challenges to the development of a smart city, namely: the escalation and complexity of the SoS to be developed, the multitude of stakeholders, the variety of domains, and emergent behaviors of the systems within. In this context, they described the challenge of the unification of the information to handle the heterogeneity and the interoperability of the system under analysis using a global meta-layer.

Semantic data integration. One popular strategy for data integration is to use knowledge representation and ontologies. In computer science, an "ontology is an explicit specification of conceptualization. The term is borrowed from philosophy, where Ontology is a systematic account of Existence" [158]. To define an ontology on the top of a domain, in computer science, a representation of the knowledge by a set of concepts within a domain and the relationships between those concepts must be provided. This approach has been implemented and described in multiple cases, like [55, 71, 326, 383]. The benefits of semantic data integration are the modularity, scalability, fast and easy integration of different formats of data while removing the need to have a centralized system to store all the data together. Bansal et al. [55] define a general ETL framework, involving the creation of the semantic data model as a basis to integrate the data coming from multiple sources. This is followed by the development of a distributed data collection that can be queried using SPARQL query language. Psyllidis et al. [327] focus on smart cities domain and present a similar approach. The data coming from multiple heterogeneous urban sources are integrated into a global ontology. On top of that, the authors define various interactive Web components (e.g. Web ontology browser and interactive knowledge graph) to access the integrated ontology graph. Bianchi et al.[68] try to combine the definition of a semantic layer with a tool that provides to domain experts the possibility to perform in autonomy the integration of multiple and heterogeneous smart city data sources. Gaur et al.[150] propose a multi-level Smart City Architecture integrating data coming from wireless sensors about pressure, temperature, electricity, and others. Their architecture is composed of four layers and each layer has one responsibility. Layer 1 receives data in many different formats. Layer 2 is in charge of processing all the data into a single format as Resource Description Framework (RDF). Layer 3 contains the inference engine for data integration and reasoning using semantic web technologies. Finally, Layer 4 is responsible for querying data. A different approach based on RDF-format data integration is presented by Consoli et al. [106]. There, the authors describe a platform implementing an ontology-integration approach that leverages on the help of domain experts. For each data source, an ontology is created. The common conceptual layer allows to convert all data in a target RDF data model. A similar solution for RDF-format data integration from sensors is presented in [384].

Bischof et al. [71] share the consensus on the effectiveness of a semantic modeling strategy for smart cities and on the conceptual data model. The approach considers the data stream annotation with descriptions for data privacy and security, and data contextualization using hierarchies to categorize smart city data. In detail, the solution is based on the definition of a semantic description for smart city data, which is heterogeneous in nature, to facilitate discovery, indexing, querying, etc. for future services. They consider data heterogeneity not just from the format point of view but also explore the nature of data considering, for example, the different units of measurement that are provided. They propose to start collecting metadata and semantic descriptions and try to find a compromise with respect to the volume that this metadata might represent. The approach ends with the definition of a Semantic Sensor Network (SSN) ontology developed by a W3C incubator group which focuses on organizing and describing sensor capabilities and data processing. The HyperCat [112] project developed a standard knowledge representation using knowledge graphs to provide a uniform and machine-readable way to discover and query data distributed among many data hubs, where each data hub can provide inputs from different IoT components and networks. In this approach, applications can identify and use the data they need independently on the specific data hub they belong to. Finally, we can also cite the CityGML open data model based on XML format that is a standard for the storage and exchange of virtual 3D city models [157].

A semantic data integration approach is of interest of the organization bodies as well. For example, it's been proposed by the AIOTI working group. Special attention must be devoted to the SAREF extension for smart cities [353] that provides a detailed model for some interesting use cases. The International Organization for Standardization [207] also works on smart city ontologies, for example, the foundation level concepts [211], the indicators [208] (populations, etc.), and the city-level concepts [211]. These ontologies constitute a very interesting and rich source for developing standardized access tools and models and have been considered in multiple approaches that follow a semantic modeling strategy.

Structural data integration. Many efforts recently consider data integration from a less abstract point of view and explore the new possibilities offered by cloud platforms or data distribution tools. This kind of data integration looks at data as small pieces that must be integrated from a structural point of view. No generic data model is provided, and no abstraction is defined at the application level. Structural data integration differs from model data integration because it does not strictly need a generic and abstract schema in a target model, unifying the global vision on data. This kind of data integration also differs from the software data integration that we will see below because it operates at the physical layer. The integration step is done in the storage layer of the platforms and frameworks. It is immediate to see that the data integration step is purely handled from a technological and structural point of view. Petrolo et al. [320] tackle the challenge of creating a smart city from the sensor standpoint. That is, they approach the problem from a bottom-up approach, and focus on the layers of data generation and consolidation. Authors propose a VITAL Platform combining the IoT and the Cloud of Things (CoT). to help alleviate the heterogeneity of data generated from different systems on a pay-as-you-go scheme. This platform combines several protocols and communication technologies, including ontologies, semantic annotations, linked data, and semantic web services to promote system interoperability. However, they mention that the challenges that still remain to be tackled are big data, privacy, and security issues. Both of these challenges have been approached by Rodrigues et al. [337] with their SMAFramework. Their framework promises to reduce the trouble of dealing with multiple heterogeneous sources (both historical and real-time generated) while allowing for multiple layers of access and security that can satisfy arising privacy and security norms. Furthermore, SMAFramework can add additional data sources in a plug-and-play fashion. Their framework is based on a Multi Aspect Graph (MAG), which they have tested on geospatial and temporal data from New York City, combining tweets with trips performed by yellow taxis. Puiu et al.[328] propose a distributed framework called CityPulse to perform knowledge discovery and reasoning over real-time IoT data streams in cities. Their architecture includes a layer called “Sensor Connection”, which is responsible for collecting the read data from the different sensors. Later, the data gathered is passed to another layer that parses it to extract relevant information. After the parsing, there is a module that performs semantic annotations by using an ontology created within the CityPulse framework. After the messages are annotated, the data is published in a message bus. Since data in the bus is already annotated with the URIs from the framework ontologies, an RDF Stream Processing (RSP) module is able to query the data over the streams. Moreover, the framework is able to discover certain events based on the analysis of the incoming annotated streams. Finally, they use a Service Oriented Architecture (SOA) in order to allow consumers to query relevant streams of the different sources or events that were discovered in the message bus.

Machine learning has also become a powerful methodology nowadays. According to research studies [126, 349], there is a synergy between machine learning and data integration and it becomes stronger over time. Modern Machine learning models help to solve the schema-matching phase that can be considered one of the hardest problem in data integration [61]. For example, Deep learning allows the comparison of long text values by their embedding representations and starts to show promising results when matching texts and dirty data. Recently, SLiMFast [334] has been proposed as a framework that expresses data fusion as a statistical learning problem over discriminatory probabilistic models and that can be adapted to explore the smart city data integration scenario. In the same context, Costa et al. [107] define a framework having a unified data warehouse that collects and stores all the available data in raw format. Their approach uses an internal model that exploits the characteristics of the Hadoop framework [33]. Unfortunately, their meta-model is not accessible from the outside and not many details about the conceptual data integration task are provided. Finally, Raghavan et al. [329] propose a prototype application based on a cloud-based API and architecture. Their solution defines specific layers providing (and restricting) simple but useful standard operations that hide the heterogeneity of the components. In these approaches, the tuning and optimization phases are critical steps that strictly depend on the characteristics of the input dataset. The challenges behind the generalization and optimization of these methodologies are just at the first exploring phase, and much interest is rising in the database research community [146, 394].

Software-delegating data integration. During the last few years, a new category of data integration approaches has been developed leveraging the power and the flexibility of the data access software layers available on cloud-computing platforms and architectures. We classify these approaches under the name of software-delegating data integration. Specifically, this kind of data integration is performed by using the various services that are provided by the cloud platforms [127]. For example, Ribeiro et al. [336] propose an architecture based on microservices developed on the top of the Hadoop framework. Their proposal is implementing and improving the approach presented in InterSCity [118] with a more scalable objective. An approach also based on distributed architecture is described in [362]. In the proposed approach, data are collected from heterogeneous sources, converted internally in a target model according to a common protocol, and made available for the target analysis. This approach can be used in any context and can be exploited also by smart city applications. A similar scenario is also present [111] where a data integrator component is in charge of dispatch requests to data sources. Software-delegating data integration is very flexible and allows quick access and integration of data according to standard operations and patterns. On the other hand, the integration possibilities and the global maintenance become fully dependent on tools and operations offered by specific platforms and offered APIs. Any change and evolution in the APIs can change the result and impact the data access.

4.2.3 . Data management

In recent years, the data has established significant propulsion with the evolution of smart cities; therefore, data management at such a scale brings challenges [49, 273]. Big data tools and technologies now support data acquisition, storage, analysis, and governance [49]. However, given the volume, heterogeneity, and distributed environment nature of smart cities, it is still difficult to integrate and manage smart city data [310]. This section will explore the challenges and state-of-the-art solutions for data acquisition, integration, storage, analysis, and governance.

Data Acquisition. Data collection or acquisition means retrieval of the data from the data sources and feeding this data into the analytics platform for storage and further processing [399]. Data in smart cities is generated by diverse sources such as IoT, economic platforms, government offices, transportation, and social media [7, 262]. These data vary greatly in their nature (text/images/video/numeric), velocities, and formats. Some data sources are quite *static*, that is, they do not change often, like geospatial map data. Some data sources provide data at regular long-enough intervals, like daily or monthly. Often, such static data sources have defined Application Programming Interfaces (APIs) to get the data, or data could be downloaded from other storage solutions. Since such data does not need to be processed and analysed immediately, it can be loaded to the data analysis platform, integrated with other data sources, and made available for deeper offline analysis (so-called batch processing) [234, 399].

Many data sources generate data *continuously and at a high frequency*, like sensor readings. Often, such data needs to be processed as it becomes available, to react quickly or detect certain pattern or anomaly. Such incrementally available data are referred to as a stream, the data record as an event, and the near-real-time processing of data as stream processing [399, 234]. In data stream terminology, we have producers (generate event) and consumers (process event) [234]. Collecting and processing streaming data requires dealing with the delayed, missing, or out-of-order data; managing situations where producers send messages at a faster rate than consumers can process; ensuring fault tolerance [234, 381, 399]. This also means that streaming data requires loosely coupled communication schemes. Common approaches here include messaging systems [234] that implement different communication patterns. For example, in a request-reply pattern, the client expects a reply from the server. In a publish-subscribe pattern, clients subscribe to certain messages published by the server that they are interested in. In a pipeline pattern, producers push the results, assuming that consumers are pulling for these [405, 399]. Message-queuing systems facilitate communication between producers and consumers via inserting and reading the messages in the queues [405, 399]. Such an approach provides loose coupling in time, solving a number of challenges of streaming systems, like lag in the capabilities to process events. Another issue is to handle the heterogeneity of producers and consumers. Message-queuing systems treat this by message brokers, namely application-level gateways that convert incoming messages to the ones that recipients can understand [405, 399]. For example, in a publish-subscribe pattern, the brokers match the topics subscribed by the consumers to the topics published by producers [234, 436, 399]. Examples of such systems are Apache ActiveMQ [28] and Apache Kafka [35].

The recent developments in big data and smart cities have given birth to a number of reliable, fault-tolerant and flexible data acquisition and ingestion solutions, like Apache Flume [31], Apache Spark [39], Apache Kafka [35], Apache Flink [30], Apache NiFi [36]. Each of these frameworks is being widely used in academia and industry depending upon the requirements. In some cases, only one framework can suffice the requirements, whereas the combination of these frameworks has also been observed [273, 310]. Therefore, while choosing any of such frameworks, one needs to be heedful of the final requirements. For example, if the data is being collected at its origin, it may require initial transformation and cleaning. In addition, as the data sources can have diverse acquisition frequencies and can require frameworks with capabilities of handling low-latency and batch-oriented data alongside data cleaning and data transformation functionalities.

Data Storage. The amount of connected IoT devices worldwide is expected to reach 50 billion [347]. Since data is a key ingredient for smart city services, solutions and tools for efficient data storage and access are needed [81, 122].

Generally, smart city applications can be considered to be data-intensive ones. In addition to application-specific requirements, such applications should ensure that data is stored reliably and available for later use, search, and processing, results of expensive operations should be saved for speedy retrieval [234].

In recent years, a number of advanced SQL, document, graph, NoSQL, NewSQL, and Big Data data storage systems have been proposed and adopted by researchers and engineers. It is clear that some of them work better for certain tasks, provide certain guarantees, and the choice is always made based on the data model and system requirements [81, 178, 234]. Examples are MongoDB [279] which is a widely used document database, Apache Cassandra [29] as a representative of wide-column data storage solutions, or VoltDB [415] as a representative of NewSQL databases. Modern storage solutions enable distributed storage and processing by utilizing replication and sharding; they provide data querying capabilities and interfaces for most commonly used programming languages and third-party systems; and cluster management functionality. Distributed implementation enables scalability, fault tolerance, and latency reduction. However, as CAP theorem says, "in a distributed database system, you can have at most only two of Consistency, Availability, and Partition tolerance" [178]. Here, Consistency refers to the property to deliver every user of the database an identical data view at any given instant; Availability promises an operational state in the event of failure; and Partition tolerance ensures the ability to maintain operations in the case of the network's failing between segments of the distributed system [178]. Therefore, in distributed implementations, usually, there is a tradeoff between consistency guarantees and other features.

Off-the-shelf big data management and processing platforms are available, like Apache Hadoop [33] and the High-Performance Computing Cluster (HPCC). Systems platform [196]. Such platforms and the software ecosystem of applications developed around them provide complete solutions from data acquisition to data storage, analysis, and results delivery to the end user. *Apache Hadoop* is an open-source Java-based framework developed for data storage and processing in a distributed environment on commodity hardware. The main components of Apache Hadoop are: Hadoop Distributed File System (HDFS): A distributed file system facilitating storage and high-throughput access to massive-scale data; Hadoop YARN: a cluster resource management framework; Hadoop MapReduce: a system for parallel processing of data; and Hadoop Common: common utilities supporting other modules [33]. In addition, a number of tools were developed for different purposes, e.g. to efficiently load the data to HDFS (like Apache Flume [31]), facilitate data storage and access (like Apache HBase [34]) process and analyze the data (like Apache Flink [30], Apache Spark [39]), maintain configuration (Apache Zookeeper [42]).

HPCC System platform is an open-source data lake platform supporting different data workflow capabilities [347]. Its main components are: Enterprise Control Language (ECL) - a data-oriented declarative programming language; Thor - a bulk data processing cluster that cleans, standardizes, and indexes inbound data; Roxie - a real-time API/Query cluster for querying data after refinement by Thor [197]. It also uses a distributed file system (DFS) for storing data in the cluster following a record-oriented approach [274]. The indexed data available in Thor clusters can be used for low-latency querying by copying in Roxie clusters, which has been specifically designed for getting much faster results, unlike Thor Cluster with batch orientation [274, 347]. In addition, as in Apache Hadoop, data is collected using different data acquisition frameworks like Apache Flume [31]. Whereas in HPCC Thor, simply a web service can be used for uploading data to Thor clusters [315].

Also, a number of big data storage solutions are proposed based on the experience and improving the challenges of big data platforms. For instance, *Apache Ozone* [37] is a scalable, robust, distributed object store for big data applications. It is designed to handle large amounts of data consistently, providing HTTP interfaces for integration with third-party applications. Ozone is built on top of the existing Hadoop components, such as YARN, HDFS, and KMS, and leverages their capabilities and integrations [312]. Ozone is also compatible with the existing Hadoop ecosystem, such as MapReduce, Spark, Hive, and Impala, and can be deployed alongside HDFS or as a standalone storage system. Apache Ozone in comparison with HDFS has several benefits. For example, Hadoop Distributed File System (HDFS) has a single namespace that can become a major challenge for metadata operations. It does not support object-based protocols, such as S3 [418], commonly used in cloud-native applications these days. Moreover, the fixed block size in HDFS can lead to inefficient storage space utilization and network overhead when it comes to small files. Apache Ozone supports multiple protocols, such as S3, HCFS, and OFS, that cater to different application needs and preferences. Ozone also provides a rich set of features, such as security, replication, fault tolerance, and monitoring [418]. The fault-tolerance of Ozone is ensured through its self-healing properties that allow it to recover from sudden node failures, making the data highly available. In addition, it is capable of supporting hierarchical namespace, enabling the maintenance of data in multiple buckets and directories [37].

Smart city services often need to analyze patterns of moving entities changing their location in time (like vehicles or mobile phone users) or extent as well (like the spread of epidemic disease) [162]. Such time-dependent geometries are called moving objects [162], therefore, storage solutions should be equipped with the opportunities to represent and query the dynamics of such kind of data. Ilarri et al. [200] categorize state-of-the-art support for moving objects into two categories: Moving Object Databases and data streams. However, they do emphasize that the boundary between these two groups is not always clear. Moving Object Databases enhance the database technologies with representation and management of moving objects [162, 200]. When compared to early spatio-temporal databases, Moving Object Databases also allow for tracking continuous change [162]. In particular, lots of research towards models to track moving objects and corresponding query languages, handling uncertainty, indexing ensuring a low update overhead and efficient retrieval of the objects is conducted, please refer to [200] for details. Prominent examples of MODs that are in active development are MobilityDB [442], extending PostgreSQL and PostGIS with the moving object support, and SECONDO [295], an extensible database management system supporting various data models. The development of big data technologies facilitated the storage and processing of traces of a large number of moving objects. A number of efforts exist nowadays to work with spatial and spatiotemporal big data [406]. Starting from equipping Apache Hadoop with support for spatial data, like data formats, spatial index structures, spatial operations (SpatialHadoop [165]), and spatio-temporal capabilities (ST-Hadoop [166]). To more recent proposals enriching Apache Spark [39] and distributed storage products with spatial or spatiotemporal capabilities. For instance, Apache Sedona [38], extending Apache Spark [39] and Apache Flink [30] with a set of tools for working with spatial large-scale data in cluster computing environments. Beast [134] is a Spark-based solution for exploratory data analysis on spatio-temporal data supporting a variety of data formats. GeoMesa [151] provides a set of tools for large geospatial data analytics. For instance, it adds spatio-temporal indexing on top of Accumulo, HBase[34], and Cassandra[29] databases to store spatial data types like point, line, and polygon. Stream processing is enabled there by having spatial semantics on top of Apache Kafka [35].

Graph databases enable efficient storage and processing of the graph data models, which is often met in the smart city domain, e.g. road network. Graph data model handles well varying granularity and hierarchical differences in data; enables evolvability, meaning that graph can be extended to reflect the changes in the application domain [178]. Examples of solutions available to help store and work with graph data models in a largely distributed environment are Neo4j Graph Data Platform [292] and Apache Giraph [32] processing system. Such solutions enable deploying graph data models on large clusters, if needed, and enable distributed graph processing by partitioning the data and processes between the nodes.

Data Processing. Most of the smart city applications rely on processing a large amount of data [404]. Depending on the application's requirements, this processing can be roughly divided into two groups: batch processing and stream processing.

Batch processing, often also called offline processing, takes a large amount of input data, runs a job to process it, and produces the output [234]. It is clear that jobs in batch processing could take a while. Therefore, they are often scheduled to run periodically, like once a day. If we go to the big data landscape of methods and technologies, then MapReduce programming model [120], allowing processing of a large amount of data in a distributed manner, was the most popular approach, implemented also in Apache Hadoop Framework [33]. MapReduce job consists of Map and Reduce tasks. First, the input data is split into portions that are processed by map tasks in a parallel manner. Then, the results of Map tasks are used by the Reduce tasks to compute the final output. It is also common for MapReduce jobs to be chained together into workflows so that the output of one job becomes the input to the next job [234]. However, the Hadoop MapReduce framework, e.g., does not have direct support for workflows, so the chaining occurs explicitly via storing intermediate results in the file system. This has certain downsides, like a waste of storage space when intermediate results get replicated, redundancy of some programming code in map tasks, and the inability to start for the subsequent tasks before the previous ones were completed [234]. Dataflow engines have been developed that aim to solve these issues. They handle an entire workflow as one job rather than breaking it up into independent subjobs. Examples include Apache Flink [30], Apache Spark [39], Apache Tez [41].

Stream processing, also often called near-real-time processing, processes events shortly after they happen. Therefore, stream processing has lower delays. There are a number of cases, when stream processing is required, like anomaly detection, finding patterns, or simply streaming analytics. Basic terminology and technologies required to get stream data to processing engines were already presented in the previous subsection 4.2.3. Here, we'll cover approaches for stream processing. Generally, there are two ways to process stream data: one-at-a-time and micro-batching [234]. For example, Apache Spark allows the use of a micro-batching approach [39]. In this approach, the processing engine splits the input data into small micro-batches, processes them, and produces the micro-batches of the results. The one-at-a-time approach is implemented by Apache Storm [40], for example.

Smart city applications are complex constructs fueled by diverse kinds of data. Therefore, hybrid approaches, combining both batch and stream processing, are often required. A number of architectural solutions to combine batch and stream processing were suggested [116]. For instance, Lambda architecture incorporates layers for batch processing, speed layer for computation on recent data (realtime views), serving layer which is specialized distributed database allowing doing queries for batch analysis results (batch views). The query result is composed of both batch and realtime views [269]. Another approach is Kappa architecture [240], which simplifies Lambda architecture by removing the batch layer. This architecture relies on the use of a log-based system (e.g. Apache Kafka) able to retain all the data that may be reprocessed if needed. Then, we need to deal only with one type of system and making changes equals just to running the new instance of the job on the whole data, writing results into a new table and redirecting the application to read the results from this new table. The old job and old results table could be stopped and removed. Liquid architecture [141] allows incorporating an incremental processing, therefore, no recomputation from the scratch is needed. Davoudian and Liu [116] discuss these and some other data system architectures (incorporating, e.g. Semantic Web technologies).

4.2.4 . Energy Conversion

Energy is an important component in any Smart City context, and I also conducted a study to investigate data sources in Energy Conversion algorithms. The study did not only compare the approaches from a data point of view but also wanted to analyse the Artificial Intelligence algorithms already explored in order to better understand how data and AI can be aligned in the field.

Next, the most utilized AI and machine learning algorithms in energy conversion are explored. As Figure 4.4 shows, and as it has been shown in previous sections, the standard artificial neural network is the most popular algorithm, followed by the adaptive network-based fuzzy inference system (ANFIS) algorithm, genetic algorithm, long short-term memory (LSTM), recurrent neural network (RNN), Q-learning, and convolutional neural network (CNN). These top algorithms are for the most part deep learning algorithms, for the exception of genetic algorithms and Q-learning, which are optimization and reinforcement learning algorithms, respectively. Their popularity is not surprising as they offer a remarkably high accuracy and low error rates, as well as vast overall improvement to traditional methods.

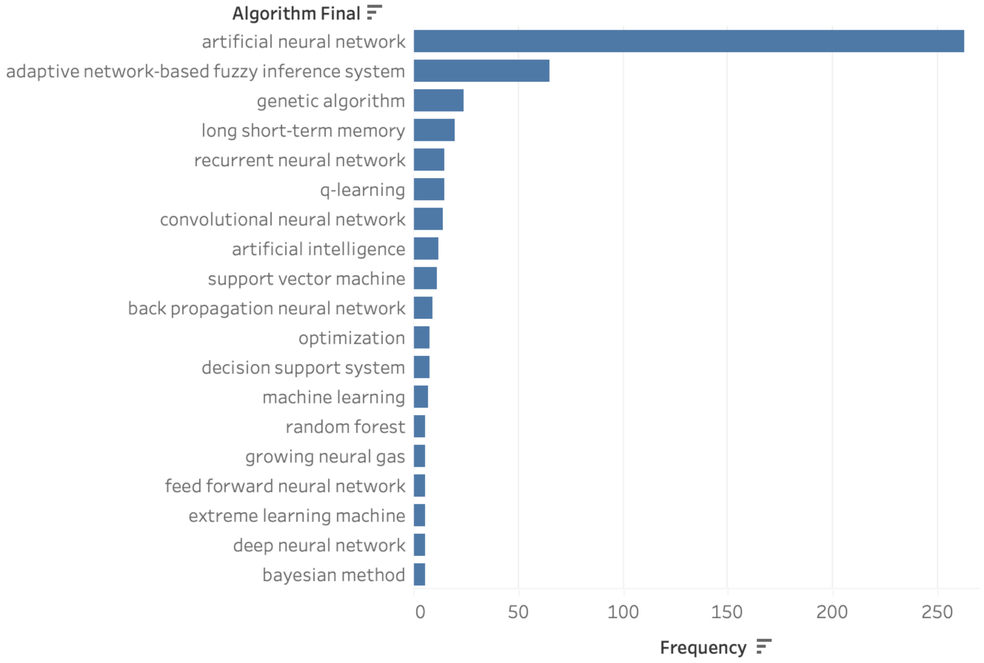


Figure 4.4: Most popular algorithms from 1994-2022.

Shown in Figure 4.5 are the most applied algorithms in recent years. The emergence of LSTM and CNN algorithms is evident. Moreover, more complex forms of reinforcement learning, such as deep deterministic policy gradient (DDPG) algorithms, are being applied. Interestingly, machine learning algo-

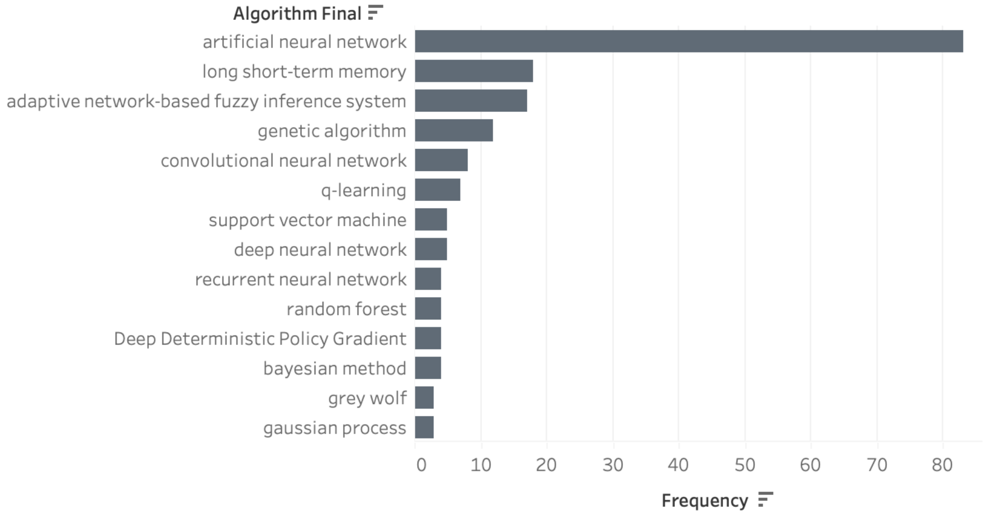


Figure 4.5: Most popular algorithms from 2020-2022.

rithms such as support vector machine (SVM), random forest, and Gaussian process regression are being explored. Machine learning offers great computation speed and little memory requirement with similar accuracy to its deep learning counterparts.

4.2.5 . Publications with Simulated vs. Real Data

Of the 224 papers with simulated data, only 97 reported data size, 115 reported a benchmarked performance measure, and 57 reported both.

Table 4.6: Comparison of Data Size as Number of Samples between Simulated Data and Real Data.

Data Size (Number of samples)		
Statistic Measure	Simulated Data	Real Data
Median	1,000	1,280
Mean	16,0198	18,461
Std.	3,547	3,518
Maximum	10,000,001	629,873
Minimum	15	14

Moreover, the distribution of data sizes in publications with simulated data (as shown in Fig. 4.6 and detailed in Table 4.6) has high outliers in comparison to publications with real data, especially when noting the maximum data size for simulated works was 10 million samples. Both sets seem to have similar medians, means, and deviations, surprisingly, with the real data set having on average 2,000 more samples.

Interestingly, the distribution of benchmarked performance measures in pub-

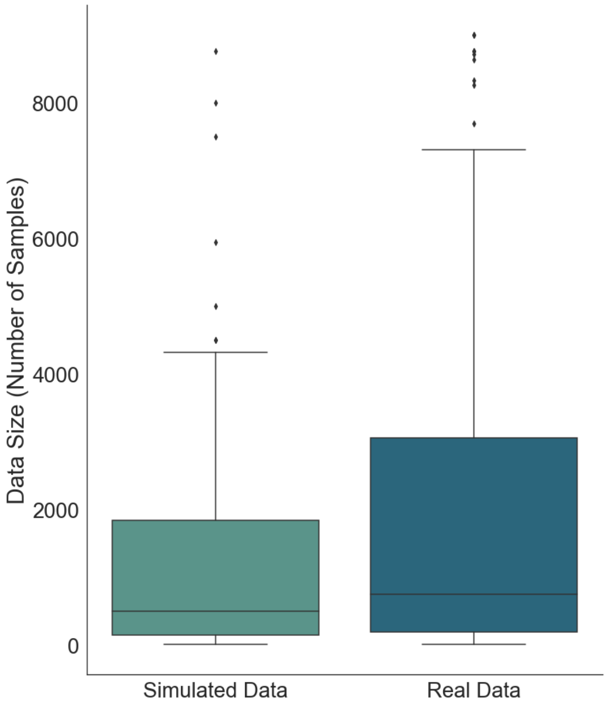


Figure 4.6: Comparison of data size distribution between simulated data (green) and real data (blue).

lications with real data (as shown in Fig. 4.7 and detailed in Table 4.7) has higher outliers than publications with simulated data, especially when noting the maximum for improvement provided by a publication using real data was 92 times the status quo. Overall, it seems that studies with real data manage to provide a greater percentage improvement than simulated data studies.

Benchmarked Performance Measure (% improvement)		
Statistic Measure	Simulated Data	Real Data
Median	20.0	23.8
Mean	108.6	556.4
Std.	63.8	54.5
Maximum	7,580	92,000
Minimum	0.06	0.2745

Table 4.7: Comparison of Benchmarked Performance Measure as Percentage Improvement between Simulated Data and Real Data.

4.3 . Related work

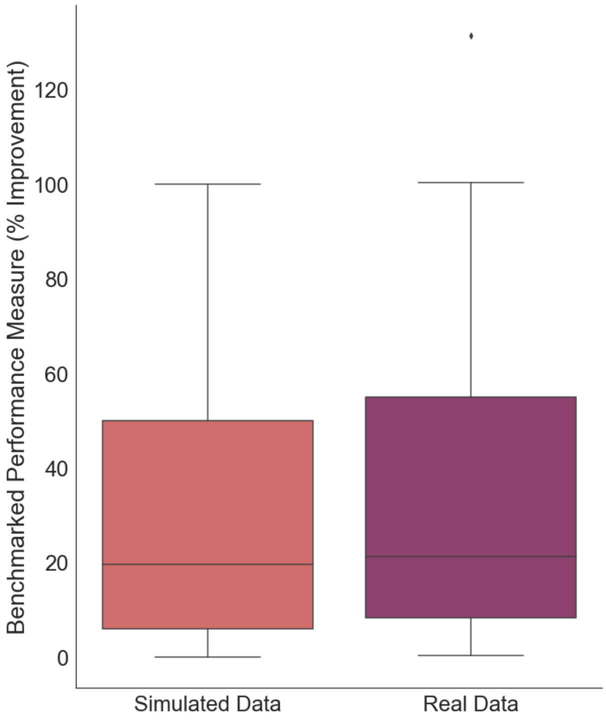


Figure 4.7: Comparison of benchmarked performance measure distribution between simulated data (pink) and real data (purple).

Urbanization and development of cities provide vibrant opportunities for academia and industry, which inspire a number of significant related research. For instance, Kitchin [231] provides a constructive view on the overall types of big data and smart urbanism. He also stresses the very relevant challenge of the corporatization of city governance and a technological lock-in when all the smart city-associated methods and technologies are available to large software and hardware companies, seeing this as a potential market for their products.

A number of research articles address the technological challenges for smart cities. Santana *et al.* [350] analyze requirements and software platforms for smart cities based on 23 projects. Authors placed these into four categories, including Cyber-Physical Systems, Internet of Things (IoT), Big Data, and Cloud Computing. Functional and non-functional requirements for smart city software platforms have been carefully investigated. Habibzadeh *et al.* [164] explores challenges, requirements, and solutions for sensing, communication, and security planes of smart cities. Similarly, Chamoso *et al.* [91] review technologies used for smart city developments, as well as propose their own solution for global architecture for service management in smart cities. Edge and fog computing paradigms offer promising solutions for smart cities. For instance, Perera *et al.* [319] explore the opportunities of fog computing for sustainable smart cities. Khan *et al.* [228] reviews edge computing applications in smart cities. Authors propose an edge computing taxonomy for edge computing-enabled smart cities, where the main blocks include security, edge analytics, edge intelligence, resources, caching, resource management, characteristics, and sustainability. Perera da Silva *et al.* [111] explores fog computing platforms published by the research community between 2015 and February 2021. They analyse requirements for such systems, their architectural aspects, and how they support services provided to the users.

Particularly, technological issues of big data in smart cities are also covered in a few related works. Al Nuaimi *et al.* [7] review applications of big data in smart cities with the focus on opportunities and challenges for utilizing big data in smart cities. Hasehem *et al.*[179] talk about the role of big data for sustainability and improvement of living standards in cities with the focus on state-of-the-art technologies. Bibri and Krogstie [70] review the core enabling technologies of big data analytics and context-aware computing as ecosystems in relation to smart sustainable cities. Lim *et al.* [252] discuss diverse aspects of smart cities, reference models, and corresponding challenges.

A number of recent surveys address different emerging aspects of the data in smart cities. For instance, Gharaibeh *et al.* [152] provide an overview of data management issues, as well as discuss privacy and security challenges. Usman *et al.* [404] explore the collection and analysis of multimedia data produced by smart cities. The authors focus on transportation, healthcare, and surveillance use cases and discuss various machine learning algorithms that could be utilized for such an analysis. Similarly, Habibzadeh *et al.*[163] focus on application and data planes for smart city system design. The authors highlight cloud- and edge-based architectures to store and process the data, as well as describe various data analysis algorithms. Ma *et al.*[262] review the data sets being collected across 14 smart cities and the state-of-the-art in decision-making methodologies. This chapter further highlights both data and decision-making issues. Moustaka *et al.* [281] conduct a systematic review on the way how urban data is produced, collected, stored, mined, and visualized in smart cities, covering the period 1996 - 2017. Based on this review, a set of taxonomies is proposed covering the smart city data entities and methods. Some works focus more on data analysis and applications in smart cities. For instance, Chen *et al.*[95] explore the latest research on deep learning in smart cities. Authors study the problem from two perspectives, i.e. the technique-oriented perspective reviews deep learning models, while the application-oriented perspective studies representative application domains in smart cities. Finally, Deng *et al.*[123] are interested in how urban information can be visualized. The authors review urban visual analytics studies and specify 22 visualization types within spatial, temporal, and other property visualization categories.

Work	Focus	Architecture/ Platform	Data availability	Data heterogeneity	Data management	Data analysis	Privacy	Security	Ethics
[350]	requirements and software platforms	✓(ET, platforms, reference architecture)	○	○	○	○	○	○	×
[91]	technologies for SC development	✓(architecture, ET)	○	×	○ (storage)	○ (big data)	×	○	×
[319]	fog computing solutions for SC	✓(device management, commun. protocols)	✓(sensor data in fog computing)	○ (context, semantic annotation)	○ (general)	○ (fog computing)	○	✓(fog computing)	×
[228]	edge computing applications	✓(high-level edge-enabled SC, requirements, open challenges)	×	○ (context-awareness)	○	○ (edge analytics and intelligence)	○ (edge computing)	✓(edge computing)	×
[7]	big data	×	○ (data sources, quality, sharing)	×	○ (big data)	○ (big data processing platforms, algorithms)	○	○	○
[70]	big data, context-aware computing	×	✓(sensing)	×	○ (big data)	✓(big data, urban context)	○	○	×
[179]	big data	✓(big data)	×	×	○ (big data)	○	○	×	×
[252]	reference models	✓(big data)	○ (main sources of big data)	○	○	×	○	×	×
[164]	sensing, communication, and security	✓	✓(sensing, communication)	×	○	○	×	✓(crypto-, system-level)	×
[111]	fog computing platforms	✓(requirements, architecture, services)	○	○	○ (ingestion, processing, storage, query)	○	×	○	×
[152]	data management, security, ET	×	×	×	✓(acquisition, coord. & management, quality & integrity, cloud vs fog, dissemination, ET)	✓(ML, DL, real-time analytics)	○	✓	×
[404]	big multimedia data in SC	×	×	×	✓(multimedia data collection platforms)	✓(representation learning algorithms, DL, data analytics)	×	×	×
[163]	data, applications planes of SC	✓	×	×	✓(requirements, architecture (cloud, edge), storage & processing)	✓(data analytics, ML, DL, visualization)	×	○	×
[95]	DL in SC	✓	×	✓(sensor, image/video, text)	×	✓(DL, applcations, challenges)	○	×	×
[262]	data sets, decision making	×	✓	○	○	✓(modeling, decision-making)	○	○	×
[281]	data analytics, SLR	○ (SC as a data engine)	✓(urban data taxonomy)	×	×	×	✓(data analytics taxonomy)	×	×
This work	data challenges	✓(architectures and platforms)	✓(open, citizen-contributed, commercial, private-public partnership)	✓(model, semantic, structural, software-delegating)	✓(acquisition, storage, processing, governance)	✓(trustworthiness, technological, methodological, ethics)	✓	✓security (in-transit, at-rest, in-proc.)	✓

SC - smart city, SLR - systematic literature review, DL- deep learning, ML - machine learning, ET- enabling technologies
✓- comprehensive coverage, ○ - some discussion, × - not discussed or very light mention

Table 4.8: Existing surveys about smart city and their coverage of topics presented in this chapter.

Recently, more aspects related to data privacy and security are covered. For example, Eckhoff and Wagner [130] provide a taxonomy of the application areas, enabling technologies, privacy types, attackers, and data sources for the attacks in smart cities. Based on that, state-of-the-art privacy-enhancing technologies are reviewed and future research directions are discussed. Similarly, Sookhak *et al.* [370] look for the taxonomy of security and privacy issues of smart cities, highlight the security requirements for smart cities, explore state-of-the-art of security and privacy solutions, and present open research issues.

Finally, emerging concepts of digital twins, metaverse, and metacities attract research interests from academia. For instance, Mylonas *et al.* [282] explore the digital twins landscape in the context of smart cities. In addition to studying the domains where digital twins are presented, the authors also emphasize some challenges related to data from digital twins perspective. Similarly, Bibri *et al.*[135] explore the emerging trends enabling data-driven smart cities for the digital and computing processes framework underlying the Metaverse as a virtual form of data-driven smart cities.

When compared to existing surveys, this review chapter is based on two works that focus on the data integration aspects of smart cities and energy conversion. We provide up-to-date state-of-the-art understanding of what the smart city is, how "smartness" can be measured, how energy conversion can be supported by data, and what the data challenges are in these domains.

4.4 . Conclusion

This chapter covered multiple aspects related to data, smart cities, and energy conversion, focusing on the ones related to data management, showing how data integration and data for artificial intelligence is still an open research challenge. Further research is needed to understand how to measure the smartness of the city since it is not so simple. Indicators, if any, should be considered carefully, namely what kind of, how to measure and assess the quality of measurement, and how to interpret it. Moreover, cities should be evaluated individually, considering their own cultural and historical circumstances, development goals, and progress.

However, the key challenge is still in data. How the data can be used securely, how the data can be shared, how it can be ensured that the data is used according to the claimed specifications, how to ensure the data quality, how to ensure proper data representations, and there are many more questions. These issues are easy to address when dealing with a single individual system. However, it is challenging to achieve this kind of proper data pipeline in a large-scale ecosystem comprising a number of data providers, data processors, and services.

The smart city domain is quite unique in the variety of data used for the services provided. Therefore, addressing data heterogeneity issues is of utmost importance. We have inspected related research, which we have categorized into model, semantic, structural, and software-delegating data integration. Each approach has its own advantages and drawbacks, discussed in the corresponding subsection 4.2.2.

We have also seen how the application of AI techniques in energy conversion shows exponential growth and opportunities for the future. Overall, research in this field that uses real data in comparison to simulated data is published in a 3:2 ratio, respectively. As expected, publications that use real data, as opposed to simulated, tend to show overall greater performance by their algorithms in comparison to the benchmarks; they also tend to have a larger number of data samples. The authors recommend that papers adopt standard and explicit practices of data size, accuracy, or error rate, and benchmark performance reporting - by doing so, other researchers can understand the implications of the findings and adopt and implement favorable algorithms accordingly.

All these analyses cannot be performed without data, then data integration is still a key enabler for smart city services and energy conversion.

5 - Time series for AI

Integrating data from heterogeneous sources has always been a topic of interest for the data research community. Over the last years, the challenge has been oriented to the objective of integrating data for feeding the Artificial Intelligence Algorithms and also optimizing their execution. In this chapter, we first propose PROCLAIM (PROfile-based Cluster-Labeling for AttrIbute Matching), a metamodel that performs an automatic, unsupervised clustering-based approach to match attributes of a large number of heterogeneous sources. Then we introduce GeoTS, a Python library to apply cutting-edge time series classification models to perform well in correlation in a completely automated setting, on top of our data. As input, we take the drilling trajectory depth and gamma-ray well logs, which measure the natural radioactivity across the well depth trajectory. The top depths of the formations are predicted as an output.

The chapter is adapted from the following papers:

- Molood Arman, Sylvain Wlodarczyk, Nacéra Bennacer Seghouani, Francesca Bugiotti - *PROCLAIM: An Unsupervised Approach to Discover Domain-Specific Attribute Matchings from Heterogeneous Sources*. CAiSE Forum 2020
- René Gómez Londoño, Sylvain Wlodarczyk, Molood Arman, Francesca Bugiotti, Nacéra Bennacer Seghouani- *Weakly Supervised Named Entity Recognition for Carbon Storage Using Deep Neural Networks*, DS 2022
- Shwetha Salimath, Sylvain Wlodarczyk et Francesca Bugiotti, *GeoX: Explainable neural network for time series classification, a geoscience case study*, KDD 2025

The chapter is organized as follows: Section 5.12 reviews the related studies on schema matching and the available tools. Section 5.1 presents a brief overview of PROCLAIM. Sections 5.4, 5.5 and 5.6 detail each building block of PROCLAIM. Section 5.7 illustrates the results of our experiments in two different domains for the production of PROCLAIM metamode. Moving to the AI models used in the domain to Section 5.8, describes the dataset used for evaluating GeoTS and defines evaluation metrics, Section 5.9 illustrates the clustering algorithms and the data cleaning process. Section 5.10 explains the methodology and compares our models with the baseline model. Section 5.12, presents the state of the art in the domain of time series analysis. We discuss the results of the experiments conducted and the model explainability in Section 5.11 and we conclude the chapter in Section 5.13.

5.1 Introduction 114
5.2 Time Series and AI 115

5.3 PROCLAIM Overview 116
5.4 Preliminary Phase 118
5.5 Attribute Profile Representation 119
5.6 Attribute Labeling 122
5.7 Experiment Results 124
5.8 Dataset and evaluation metrics for AI 126
5.9 Data cleaning 129
5.10 Classification operation 130
5.11 Experiments and Results 133
5.12 Related Work 138
5.13 Conclusions 140

5.1 . Introduction

During the last years, the availability of multiple and heterogeneous data sources has given new perspectives to the schema matching problem which is a fundamental step for data integration. A large number of research works exist in the literature, the main task in these approaches is to identify the correlation between the attributes using dataset values, semantic and syntactic rules to detect the correspondence between attributes during the schema matching process [15]. Most of the works on schema integration assumed a global (mediated) schema and then tried to find a solution for better matching on mostly a pairwise matching between the source schema and the mediated schema. In this context it is very difficult to define a global schema that matches all the attributes of a given domain [216]. Moreover, real-world data is always noisy, and for most of the integration methods, data cleaning is needed. However, in terms of big data, data cleaning is expensive and time-consuming. In this chapter we develop a heuristic method that can deal with real-world and massive data.

In this chapter, we present PROCLAIM (PROfile-based Cluster-Labeling for AttrIbute Matching), an unsupervised method for matching attributes coming from a large number and heterogeneous sources in a specific domain. Our results show that PROCLAIM is an effective fully automatic method to discover a set of meaningful vocabularies which are the backbone of the definition of a specific domain. PROCLAIM defines the concept of attribute profile by taking into account the data type using: (i) the statistical distribution and the dimension of the attribute's values, and (ii) the name and textual descriptions of the attribute. These properties give a unified representation to each attribute.

The cluster-labeling function takes as input these properties to automatically assign a set of labels to a high number of attributes.

One of the domains in which we can prepare data also using this technique is the study the lithography of the Earth's subsurface. The domain studies the characterization of different stratified layers called geological formations. This study performs a well correlation task to model and characterize reservoirs. This operation links the beginning of specific geological formations called tops using measurements from drilled wells.

Although data are abundant, the traditional algorithms used for well correlation are semi-automated, requiring significant time and high computational cost. This work introduces GeoTS, a Python library to apply cutting-edge time series classification models to perform well correlation in a completely automated setting. As input, take the drilling trajectory depth and gamma-ray well logs, which measure the natural radioactivity across the well depth trajectory. The top depths of the formations are predicted as an output. The gamma-ray signatures are extracted around the top depths assigned by geologists. Preprocessing is performed to clean and cluster these signatures using the Dynamic Time Wrapping (DTW) distance and and the density-based algorithm HDBSCAN Implementation of existing deep learning architectures (FCN, InceptionTime, XceptionTime, XCM, LSTM-FCN) and new architecture (LSTM-2dCNN, LSTM-XCM) are performed. Our experiments demonstrate faster computation with an increase in accuracy. Grad-CAM is used as visualization technique for model explainability. Experiments were performed using Colorado oil fields and deployed on Wyoming oil fields. The deployment has provided us with critical insights regarding the improvements needed.

5.2 . Time Series and AI

Problem The main problem is that most of the information related to geological formation is currently estimated by geologists using mud logs and the rocks extracted during borehole drilling to study their characteristics. This process is tedious and time-consuming [357]. A marker top refers to the beginning of the intersection between the well and the formation. According to domain experts, a pattern is observed at the change of the formations, known as marker signature. The marker signatures are expected to be similar across the wells, at least within a region [268]. Machine learning (ML) and dynamic time warping (DTW) algorithms [283] have been used to semi-automate the process of identifying these formations using wireline logs [139, 254]. Wireline logs are records of formation properties across the well depth measured using a variety of sensors. The nearest neighboring wells are selected through clustering. Observing these well logs, experts select the marker signature manually. DTW is used to search for the best match in the new wells. However, this method still has a high time complexity and is subject to human bias [6]. Using deep learning to extract information from wireline logs efficiently would save time and resources.

Proposed Solution To completely automate the process, we first cluster the wells according to their geological location. We then extract a sequence of formation that follows an order of occurrence. The marker signatures are clustered using the DTW distance to be able to group and identify a distinct marker pattern in the region. We approach the problem as pattern identification using time series classification (TSC) [330, 371].

The chapter documents a comprehensive framework encompassing everything from cleaning well log data to estimating the marker depth for geological formations. GeoTS allows for the comparison of state-of-the-art time series classification models such as FCN [259], LSTM [190], Inception Time [206], LSTM-FCN [223], and new models for well-correlation tasks. We have used the DTW-based algorithm as our baseline. We have implemented Grad-CAM [355] for time series, which allows us to understand the feature importance and selection for the classification task. Our framework has achieved a higher score than the industrial baseline process.

The contributions of this chapter are summarized as follows:

- Provides a new streamlined framework called GeoTS for reservoir modeling, allowing its utilization by geologists to understand the lithography of the earth's subsurface.
- Introduces new hybrid TSC algorithms and provides model explainability with Grad-CAM implementation for time series.
- Deployment on real-world dataset to compare the results of GeoTS framework.

5.3 . PROCLAIM Overview

Schema matching aims at discovering semantic correspondences of attributes of schemas across heterogeneous sources. Our goal is to get a global attribute schema for all the independently developed schemas of the same domain, which can be formalized as follows.

We consider a set of schemas $\mathcal{S}$=$\{S_1, S_2, ..., S_n\}$ and the set $\mathcal{A}$=$\{A_1, A_2, ..., A_n\}$, where A_i is the set of attributes of the schema S_i. *Schema matching* selects sets of similar *attributes* creating m different groups (G_j), as illustrated in Example 1.

A *labeling function* f_L*(G)* associates then to each group G_j a label representing the semantics of the group.

Example 1 *Consider three schemas about rental car descriptions:*
S_1= $\{$*Fuel_Type, Location, Mileage, Name, Price, Year, Transmissio*$\}$
S_2= $\{$*Country, Disp., HP, Mileage, Price, Type*$\}$
S_3= $\{$*fuel_type, maker, manufacture_year, mileage, model, price_eur, transmission*$\}$
The attributes will be matched in the following four groups:
G_1 = $\{$*Fuel_Type, fuel_type, fuel, fuelType* $\}$
G_2 = $\{$ *Location, Country, city, county_name, state_name* $\}$
G_3 =$\{$*Name, maker, brand*$\}$
G_4 =$\{$*Transmissio, transmission*$\}$
Notice that the attributes are grouped despite different spellings and semantics. Then the labels $\{$*Location, Brand, Fuel, Components,* $\dots\}$ *will be assigned to the groups by the labeling function* L. $f_L(G_1) = Location$, $f_L(G_2) = Components$, *etc.*

The main question addressed in this research is how to define an automatic process that discovers a set of labels that can effectively represent a global attribute schema for a specific domain. The PROCLAIM method is proposed as an answer to this question. PROCLAIM is a new approach that enables the automatic, holistic schema matching, which leads to constructing a global attribute schema for a specific domain. Let us illustrate the procedure by following the main steps it involves, with the help of Figure 5.1:

1. a set of heterogeneous sources with different schemas ($\boldsymbol{S}$) is provided as input;

2. the data from all sources are stored in columnar format storage;

3. the data type of each attribute is identified and data with the same type are stored in the same set ($\boldsymbol{S}_{d_K}$);

4. an attribute profile is computed based on the specificity of each data type ($\boldsymbol{S}_{d_K}$). This profile for all kinds of attributes can contain at most four properties (statistics, description, unit, and name property). The assigned profile to each attribute will be converted to a numerical vector;

5. an automatic labelling process is defined to find all similar attributes and gives a unified name to each of them. This process includes two principal components: (1) finding the most similar attributes from different schemas, (2) giving an automatic label to each attribute by a defined labeling function (L_f). A density-based clustering algorithm will be applied to the numerical profiles to find the most similar attributes. Each profile vector represents a unique attribute;

6. the list of automatically computed labels will define a global attribute schema for a specific domain.

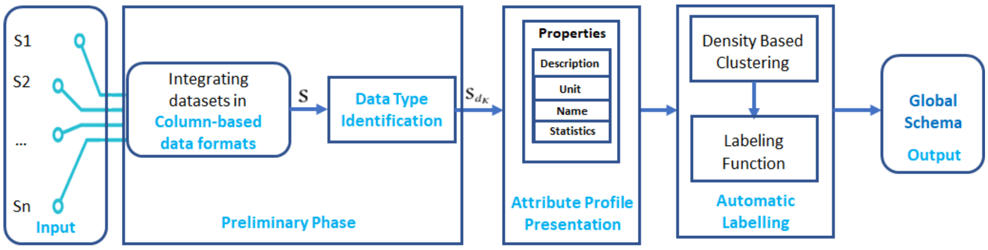


Figure 5.1: The framework of PROCLAIM to discover a global schema.

As explained in detail in the following sections, PROCLAIM can be applied on real-life noisy data. The method is designed to handle a large number of heterogeneous schemas and proposes a unified numerical profiling of information of any data type. The approach enables the usage of common machine learning algorithms such as clustering. Finally, the automatic labeling and merging of clusters allow the definition of a global schema that represents the synthesis of the heterogeneous schemas.

5.4 . Preliminary Phase

Some of the building blocks of PROCLAIM can be considered as initial steps to prepare the original datasets. Two main steps are defined as the initial steps in the preliminary phase of PROCLAIM (1) targeting data into a columnar datastore, (2) identifying the data type.

5.4.1 . Column-based data formats

Column-based data formats organize data in a set of tables. Each table contains a set of rows, and each row has a set of columns, each with a name and a value. Rows in a table are not required to have the same attributes. Data access operations are usually over individual rows and show the best performances when retrieving only a subset of the attributes of a table, when data sets are sparse and contain lots of empty values [294]. Moreover column-based data formats process big datasets efficiently since provide large-scale parallelization and effective partitioning strategies. PROCLAIM for its calculation needs a tuple for each value of attributes showing the name of the attribute and its value. In this case, storing the data in columnar-based format is much efficient.

5.4.2 . Data Type Identification

When the search space is large (the number of attributes or schemas is big), matching the complete input of schemas may require long execution times, and achieving high-quality results may be difficult. One way to reduce the search space is to find similar attributes within the same data types. The heterogeneous sources provide attributes in different data types. Since the type of the attributes may not be provided in the metadata of sources, we need to identify the types given the values. One main problem in this step is the fact that the original datasets are not clean. We will consider the type based on the data type of the majority of the instances (values), considering a standard threshold of 0.8. Here, we just consider five data types, but this set can be extended if it is necessary:

- numerical representing all attributes whose value just contains an integer or a float;
- categorical containing all strings, characters, and mix data type;
- date representing date and time such as datetime, timestamps and etc.;
- rare classifying attributes which have less than 10 instances (i.e., primary colours);
- unique referring to attributes with a unique value/cardinality equal to one (i.e., a column with a measure constant value).

5.5 . Attribute Profile Representation

Once we have all the attributes belonging to the same data type (S_g), we can group them to discover attributes coming from different schemas which contain the same information (e.g.,$\{$name, maker, brand$\}$ in our example). PROCLAIM performs clustering and labeling based on the computation of a similarity matrix of numerical profiles of attributes. Before applying our algorithm, we must convert an attribute to a numerical profile based on its data type. According to our representation, any attribute is characterized by a maximum of four components according to the data type to which it belongs. These components are description, unit, name, and statistics. In this section, we provide a description of each component of the profile and its contribution to the analysis of the attributes classified in any of the six data types introduced in the previous section. Notice that the rare type attributes are ignored due to the impossibility of computing a valid statistic.

Description Property The majority of datasets have a descriptive part for the schema where the meaning of each attribute can be found. In other cases, the description is not provided, but the used values belong to domain-specific terms or abbreviations, and this description can be retrieved, for example, using domain-specific Wikis.

To create the description profile, first of all, we remove the stop-words and then we apply the stemming method over a bag of tokens. Then, for each description, the stems and the occurrence of each term (in all the different descriptions for any specified attribute) are used to build the description profiles. Removing stop-words in a specific domain is necessary, since these words can appear in almost all descriptions and can cause false similarities (e.g., for the domain of cars, the words such as car, vehicle, automobile and etc, are the domain stop-words). We then transform the descriptions to categorical variables. Next, feature engineering is required to encode the different categories into a suitable numerical feature vector. One-hot encoding is a simple but efficient, widely-used encoding method [90]. An example of converting categorical variables for some attributes to numerical values can be seen in Table 5.1.

Attribute	displac	volum	engin	cc	repres	kw	ccm
ENGINE_DISPLACEMENT	0	0	0	0	0	0	1
ENGINE_POWER	0	0	0	0	0	1	0
DISP.	1	0	1	0	1	0	0
ENGINE	1	1	1	1	0	0	0

Table 5.1: One-hot encoding for converting descriptions to numerical feature.

Unit Property Dimensions and units are fundamental tools to explain the characterization of phenomena [343]. A dimension is a measure of a physical variable by fundamental quantities without numerical value, such as distance, time, mass, and temperature. However, a unit is a specific way to assign a measurement (with numerical value) to the dimension, e.g., a dimension is length, whereas meters or feet are relative units that describes length [343]. Dimensions and units are commonly confused, despite the fact that the solution to most problems must include units. The distribution of the same entity in different units can be shifted, but by consideration of the same dimension, the similarity of shifted distribution can be found. Also, attributes with units related to same dimension are related to each other through a conversion factor, such as Kelvin or Celsius which measures the dimension of temperature and they can convert to each other. Given a dataset, the related units can be found thanks to the descriptive part of the schema or taking into account also the instances (near the value or in a separated column). The units and their mapped dimensions of attributes can be extracted and recorded separately. In Table 5.2 we show dimensions and units characterizing some attributes of our running example. The dimension is also encoded using one-hot encoding approach.

Attribute	Unit	Dimension
ENGINE_DISPLACEMENT	CCM	VOLUME
ENGINE_POWER	KW	POWER
PRICE_EUR	EUR	PRICE
ENGINE	CC	VOLUME

Table 5.2: Some attributes with their units and associated dimension.

Name Property The name of an attribute can also be useful for the analysis. Names often contain concatenated words and abbreviations. Thus, they first need to be normalized before they are used to construct a profile to compute linguistic similarities. First tokenization is applied but it may not be enough; e.g for the name 'vehicleType', the name should be split into word 'vehicle' and 'Type'. In this regard, we compare all names of other attributes and see if one of them is part of the name string, this breakdown will be done.

Statistics Property The statistics profiles concern categorical and numerical data types. PROCLAIM uses descriptive statistical analysis to produce a profile for each attribute which not only defines the characteristics of an attribute but also enables comparing the profiles to find similarities. In the following, we list the most important statistical measurements regarding numerical and categorical data types.

- numerical data type:

  For the numerical data type, there are several measures that can be studied. The domain under analysis and the characteristics of analyzed data will help us to select the significant ones. these measures can be variability or dispersion of distribution of values per each attribute, symmetry of the distribution, the number of instances (cardinality) and central tendency.

- categorical data type:

  For the categorical data type, the considered statistics profile contains the top most frequent values among all instances of one attribute. This set of top most frequent instances can design a pattern for an attribute.

Since other components of attribute profiles are encoded using a one-hot encoding approach, we decided to apply the same method to the statistics profile. First log transform will normalize the distribution with left or right skewness, then the distribution is presented into categorical scale using binning and finally encoded. We obviously lose the numerical nature of the statistics, but we can merge this vector easily with the other vectors without a normalization issue.

Example 2 *In Table 5.2 we present the statistics profile for four numerical attributes. As a result of this analysis, we can see that the 'Engine' and 'Engine Displacement' have the same normalized distribution. Normalized data with log transformation is shown in 5.3.*

For each attribute of the dataset, we compute the global profile, which is made of the four properties described in this section. Each profile is built by considering the type of attribute, and the global profile is finally converted into a numerical vector.

Attribute	5%	25%	50%	75%	95%	Count
DISP.	90.9	113.75	144.5	180.0	302.0	32
ENGINE	993.0	1198.00	1497.0	1995.0	2982.0	101
ENGINE_DISPLACEMENT	1124.0	1400.00	1600.0	1968.0	2967.0	158
ENGINE_POWER	44.0	65.00	80.0	103.0	161.5	114

Figure 5.2: Statistics Profile.

Attribute	5%	25%	50%	75%	95%	Count
DISP.	5	5	5	5	6	3
ENGINE	7	7	7	8	8	5
ENGINE_DISPLACEMENT	7	7	7	8	8	5
ENGINE_POWER	4	4	4	5	5	5

Figure 5.3: Normalized Statistics Profile.

We finally produced a dataset that is made of a collection of vectors that will be the input for the next steps of the computation.

For each of the four properties, we propose a weighting factor on the properties that is adjusted according to the data type of the attribute. For example, for numerical and categorical variables, the attribute name can be ignored because this information is uncertain and the distribution of the values is very important.

5.6 . Attribute Labeling

The attribute labeling is a three step process that (1) performs attribute clustering, (2) assigns a label to each cluster, and (3) merges clusters having the same label. Step 3 creates each single attribute of the global schema. In this section, we are going to detail each step of the process.

5.6.1 . Clustering

The calibrated numerical vectors produced as described in Section 5.5 allow us to apply clustering to find similar groups of attributes ($G_i \in G$). PROCLAIM uses a density-based clustering method. Density-based clusters are connected, dense areas in the data space separated from each other by low density areas. Density-based clustering can be considered as a non-parametric approach, since this method makes no assumptions about the number of clusters or their distribution [93]. In higher-dimensional space, the assumption of a certain number of clusters of a given distribution is very strong and may often be violated. However, other parameters should be identified, e.g., a density threshold that is the minimum number of points (MinPts) and the radius of a neighborhood (ϵ) in the case of DBSCAN [136] and OPTICS [27]. Sparse areas, as opposed to high-density areas, are considered as outliers (noise). This results in having points in the sparse areas that are not assigned to any cluster since in general each outlier can be considered as one cluster containing just one element. As a result, 1) It is not necessary to specify the number of clusters; 2) It is not necessary that all the points belong to at least one cluster.

OPTICS [27] (Ordering PoinTs to Identify the Clustering Structure) and the aforementioned DBSCAN is a popular density-based clustering algorithm. Despite all the similarities in the core concept of both algorithms, they have fundamental differences [27]. PROCLAIM uses OPTICS. In PROCLAIM, we want to reduce the chain of core profile effect [27] in order to have small clusters with very similar profiles; hence, we set a very small value (e.g 3) for the MinPts input of OPTICS. We will then compute many clusters and have many outliers. To reduce the number of outliers, we run OPTICS a second time, again with a small value for the MinPts parameter, only on the profiles that were considered as outliers. The clusters computed during the second step will be added to the clusters computed during the first step. With these two iterations, we increase the number of clusters and reduce the outliers.

5.6.2 . Labeling Function

The labels for each cluster will be created by using the descriptions and names of all elements in each cluster. The stop words will be removed using the common linguistic stop words and the domain-specific ones. The idea is to select the most frequent words, bigram, and trigram terms appearing in the description and name of each attribute of the cluster. Then, the most frequent term will be the label of the cluster as shown in Example 3.

Example 3 *Consider* C_1 = $\{ENGINE, DISP.\}$ *as a cluster computed using the two-steps OPTICS algorithm. The descriptions gathered per each attribute are:*
$Descr\_Engine$ = *' The displacement volume of the engine in CC.'*
$Descr\_Disp.$ = *' : Represents the engine displacement of the car'*

The name profile of attributes can also be added to the descriptions: $Descr\_names$ = *{*$engine, disp\}$*.*

Furthermore, after removing the stop words, the following bag of words for each description will be generated:
BOW_Engine = *{displacement* $: 1,$ *volume* $: 1,$ *engine* $: 1,$ *cc* $: 1\}$
$BOW\_Disp.$ = *{represents* $: 1,$ *engine* $: 1,$ *displacement* $: 1,$ *car* $: 1\}$
$BOW\_names.$ = *{engine* $: 1,$ *disp* $: 1\}$

Moreover, we create a holistic bag of words by merging all the terms together associated with their total number of occurrences as follows:
BOW_total = *{engine* $: 3,$ *displacement* $: 2,$ *volume* $: 1,$ *cc* $: 1,$ *represents* $: 1\}$

By selecting the most represented term, we may produce some meaningless labels such as "displacement engine" rather than "engine displacement". To tackle this problem, we need to create a domain specific corpus and extract from it the bigrams and trigrams associated with the respective number of occurrences. This will be used to adjust and validate the labels.

Consider a created corpus in the cars domain which includes resources of glossaries, dictionaries, wikis and etc., which can easily be gathered online[1]*. Now, all combinations of the highest frequency words from BOW_total will be considered to create the bigrams and trigrams which already exist in this domain (the meaningful N-grams) with respect to term frequency in the corpus. The bigrams and trigrams selected will create a valid bag of terms. We will also add the most frequent word appearing in the corpus to this valid bag of terms. From Example 3 we have: Bag_of_terms = {engine displacement $: 2,$ displacement volume $: 1,$ engine $: 3\}$. To get the selected label, we take from the bag of terms the term with the maximum number of occurrences, with the priority first to the trigrams, then bigrams, and finally words.*
*The selected label for the cluster C_1= $\{ENGINE, DISP.\}$ is **engine displacement** even if the number of occurrences of **engine** is higher.*

After labeling each cluster, we can finally merge the clusters with the same label or labels that are synonyms (Example 4).

Example 4 *Consider C_2=$\{ENGINE\_DISPLACEMENT, ENGINE\_POWER\}$ as another cluster computed using the two-step OPTICS algorithm. The bag of words retrieved from related descriptions for these attributes are:*
$BOW\_Engine\_Displacement$ = *{ccm* $: 1\}$
BOW_Engine_Power = *{kw* $: 1\}$
BOW_names = *{engine* $: 2,$ *displacement* $: 1,$ *power* $: 1\}$
As the final result, the output is:
Bag_of_terms = *{engine displacement* $: 2,$ *engine power* $: 1,$ *engine* $: 3\}$
*The computed label is again **engine displacement**, which means that this cluster can be merged with cluster C_1 of the example 3. Then the new cluster contains the following attributes {ENGINE, DISP., ENGINE_DISPLACEMENT, ENGINE_POWER}.*

All the merged and labelled clusters generate a global schema for a specific domain. The label of different clusters in different data types can be the same, which enables us to integrate the attributes together even if their data types were assigned wrongly in Section 5.4.2. PROCLAIM helps to integrate the data from different sources and also creates a general schema which can help for integration or new sources or to populate a knowledge base in the specific domain.

5.7 . Experiment Results

In this section, we provide the experimental results on two datasets: one of them is our ongoing cars example and the second is from the oil and gas domain. The code of the experiment is implemented in Python 3.6.7. Parquet [414], a columnar datastore is used to store original datasets. Parquet is a

[1] Data from: `https://www.kaggle.com/`

Attributes	PROCLAIM labels	Annotated labels	Match
PRICE_EUR	price converted	price	1
PRICE	price converted	price	1
POWERPS	power	power	1
HP	power	power	1
WEIGHT	weight	weight	1
POSTALCODE	weight	address	0

Figure 5.4: Labeling for Car_Kaggle.

Data type	Precision	Recall	F-measure
numerical	85.7	85.7	85.7
categorical	73.0	58.8	64.2
date	100	100	1
Overall	82.5	72.7	77.3

Figure 5.5: PROCLAIM Evaluation.

free and open-source optimized column-oriented data storage developed on the Apache Hadoop ecosystem. To the best of our knowledge, there are no benchmark labeled datasets for comparing our results with another method. Therefore, for the car example, we have collected data from Kaggle challenges. For the Oil and Gas example, we use a large dataset.

Data set	Kaggle Challenge Name	#Attributes	#Descriptions	#Units	#Source records
S_1	Used Cars Price Prediction	13	11	4	1000
S_2	cars data	8	7	0	600
S_3	personal cars classified	16	11	4	1000
S_4	Craigslist Cars EDA	26	24	0	1000
S_5	Used cars database	20	12	1	1000
Sum	Car_Kaggle	70	65	9	4600

Table 5.5: Car_Kaggle Data set Information as the input for PROCLAIM.

Car_Kaggle The Car_Kaggle dataset was gathered from five different sources (S_1,..., S_5) about cars from different Kaggle challenges[1]. The global Car_Kaggle dataset, after merging different sources contain 78 original attributes: 70 of them have different names; 65 out of 70 attributes contain descriptions and just 9 out of 70 attributes have the provided unit. In Table 5.5, we provide the details of each schema. As first step, we run data type identification in order to discover the type of each attribute. Data for this dataset can be split in four different types and, as we can see the rare data type is not present. Unique attributes are discarded (6 attributes) and we compute the profile for the 64 remaining attributes (25 numerical, 35 categorical and 4 date attributes) and automatically assign label to each attribute. To be able to evaluate PROCLAIM, we manually labeled all the attributes. A subset of PROCLAIM labels and manual labels can be seen in Table 5.4. To evaluate the quality of PROCLAIM labels, we used three metrics: *precision*, *recall*, and *F-measure*. Precision is defined as the percentage of correct labels. We compared manual labels with PROCLAIM labels. If the pair (Proclaim label, Manually annotated label) matches, the label is considered as valid, as it can be seen in 5.4. Recall is the ratio of attributes with correct labels to all attributes (with or without labels). F-measure, which is the harmonic mean of the precision and recall. This result is shown in 5.4. These measures were calculated separately for each set of attributes (of each data type) and finally for the whole set of attributes. As it can be seen in 5.5, precision is showing a good quality of labels but since the number of attributes and sources are not big, we expected not very high recall, but still this recall is promising for the schema matching problem which in this research is not the main concern. The main goal is to have high-quality labels.

Oil_NorthSea dataset The North Sea Oil and Gas (Oil_NorthSea) dataset was gathered from OGA (The Oil and Gas Authority Open Data) website, which contains 43997 different sources with a total of 5260 attributes. 4713 of them have different names. The description is available for 3481 attributes, and a unit is provided for 1668 of the 4713 attributes. We apply the same approach as described in section 5.7. The number of different identified types of attributes is: 638 numerical, 631 categorical, 46 date, 574 rare, and 2824 unique attributes. Since the number of attributes is too big to be entirely manually annotated, we asked domain experts to label random set of attributes (20 labels for numerical and categorical attributes and all labels for date attributes - the number of date attributes are less than 50). We cannot calculate recall and f-measure here, since the manual labels are just provided for a subset of random labels. However, precision is calculated for these subsets for different experiments. Experiments are done for different profiles for each group of same data type attributes and the result is shown in Table 5.7. Cover data ratio measures the percentage of labeled attributes. The Covered_data ratio is showing a high percentage of considered attributes to discover the global schema. As can be seen, the precision of clusters for numerical, categorical, and date data type is over 90% which is a promising result. The global schema created from the Oil_NorthSea dataset contains 247 labeled attributes which covers 86% of the 1315 original attributes belong to the numerical, categorical, and date data types.

5.8 . Dataset and evaluation metrics for AI

This section introduces the well log dataset used in the experiments testing GeoTS (TSC framework for estimating geological formation to model carbon storage reservoirs). We also detail the definitions of evaluation metrics used to evaluate and compare model performances.

5.8.1 . Dataset

Data type	Profile	#Unlabeled Attr.	#Labeled Attr.	#Labels	Precision (%)	Covered_data (%)
Numerical	Stat.	84	554	107	58.1	86.8
	Descr.	122	516	112	93.25	80.9
	[Stat., Descr.]	53	585	128	86.4	91.6
	[Stat.,Descr.,Unit]	60	578	110	90.1	90.5
Categorical	Stat.	135	496	102	70.5	78.6
	Descr.	203	428	100	94.9	67.8
	[Stat., Descr.]	129	502	126	86.2	79.5
	[Stat.,Descr.,Unit]	121	510	130	92.1	80.8
Date	Descr.	16	30	3	100	65.2
	[Descr.,Name]	5	41	7	94.3	89.1
	[Descr.,Unit,Name [1]]	5	41	7	94.3	89.1
Total	[full profile]	186	1129	247	92.2	85.9

[1] Unit is not available for Date attributes

Table 5.7: Experiment results for different profiles subset

Well logging has two main types: (i) geological well logs, made from mud logs and rock samples, and (ii) geophysical well logs, which record physical measurements collected by instruments lowered into the drilled hole. These geophysical well logs are analyzed to identify lithologies, to differentiate between porous and nonporous rock, and to get reservoir characteristics from subsurface formations [23, 244]. This work focuses on geophysical well logs, which we will henceforth refer to as simply as well logs.

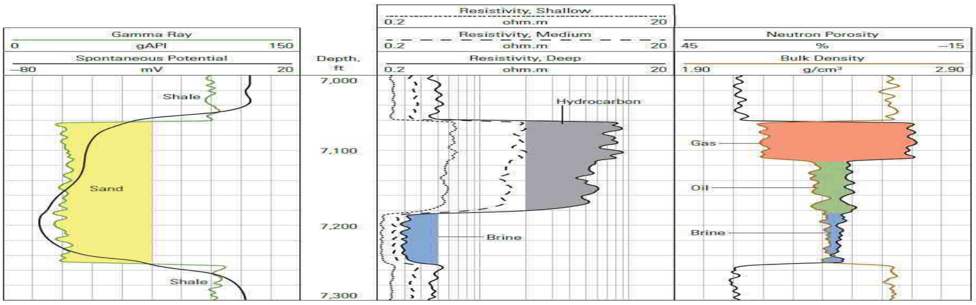


Figure 5.6: Example of gamma ray, resistivity, and neutron porosity well logs [23].

Well logs record the magnitude of a specific formation property, such as resistivity, gamma radiation, density, neutron porosity, and sonic properties, measured by the tool as it traverses an interval defined by depth. Well logs present a concise, detailed plot of formation parameters vs depth, as shown in Figure 5.6. Our study uses Gamma Rays (GR) because they are present across many wells. The GR device measures naturally occurring radioactivity from the formation due to the presence of radioactive elements, primarily potassium, uranium, and thorium [82, 186].

Colorado dataset Data were sourced from the Colorado state government website [300], where we obtained two files for each well: one with well log measurements at various depths (including GR, density, and resistivity) and another is well report containing geological formations with their marker depths.

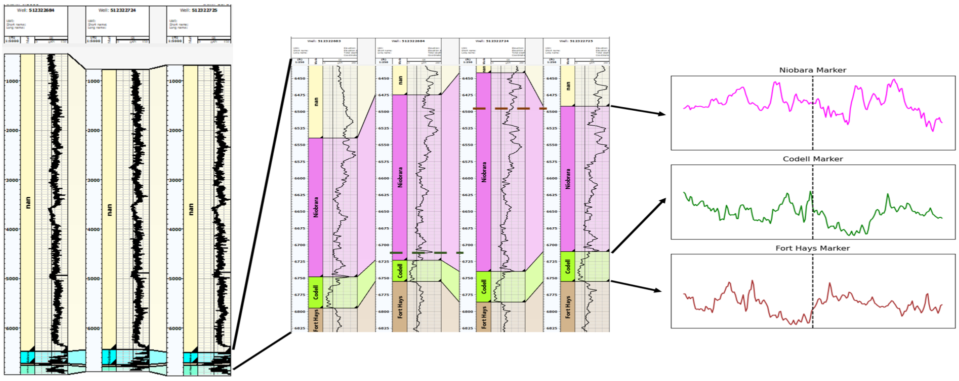


Figure 5.7: Gamma ray logs for different wells within a region.

We have used a subset of three markers in 800 wells, with log depths ranging from approximately 300 ft to 7200 ft. The markers of interest are Niobrara, Fort Hays, and Codell. Of the 800 wells, 150 are testing wells with manually verified marker depths by geophysicists due to the inherent noise in log data. The noise can be from the measuring instruments or an error in picking the top marker depth. Figure 5.7 displays the GR logs for select wells, showing overall variability and highlighting a section near the bottom that contains the formations to identify. The marker signature of these formations is also extracted and displayed. The dotted lines in Figure 5.7 indicate the correct marker depths for the marker originally assigned incorrectly.

5.8.2 . Evaluation Metrics

Marker propagation is to be able to predict the marker depth using the GR well log data and the well location. We use accuracy, Mean Absolute Error (MAE), and recall to compare model performance. All these terms are explained in this subsection.

Accuracy is computed as the proportion of labels that are correctly predicted over all of the labels for the classification model.

MAE The mean absolute error (MAE) is the absolute difference between predicted and actual depth for each marker top. MAE is then averaged over all the wells.

Recall The recall is calculated after applying a threshold T. If the absolute difference between the predicted depth $\hat{d}$ and actual depth d is less than the threshold T, we consider it a correct prediction and assign it as a true positive; otherwise, it is a false negative. The recall is then calculated per class and macro-averaged. It measures how many of the labels for a class are correctly predicted as:

$$IF(|\hat{d} - d| < T) = TP$$

5.9 . Data cleaning

The well log data set is naturally noisy due to fluid turbulence, subsurface conditions, and technical errors such as sensor noise or hardware failure. This section outlines the data cleaning and preprocessing pipeline in the GeoTS framework. Initially, missing values are addressed by imputation with averages or nearest neighbors, ensuring uniform time stamps and units.

After the initial cleaning, the GR reading around the marker depth is extracted as the marker signature with a specific window size. Geologists assign marker depths, leading to subjective bias in interpretation. In cases where the readings are wrong, the marker depth assigned needs to be corrected. Therefore, further cleaning is necessary to identify wells with clear marker signatures. Clustering with DTW distance is employed to filter wells for training the deep learning model.

Dynamic Time Warping (DTW) [283] distance is used for measuring similarity between two temporal sequences, which may vary in speed. The DTW distance is computed after realigning (warping) two time series of equal lengths. The marker signatures with a given window size are extracted, and the DTW distance matrix is calculated. The DTW distance matrix contains the DTW distances between all pairs of extracted signatures. The matrix size is $[N_m, N_m]$, where N_m is the number of signatures for particular markers in the dataset. This matrix is used to perform clustering.

5.9.1 . Clustering marker signatures

The Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) [266] was used for clustering the signatures [251]. The Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) [266] is used for clustering the signatures. It clusters based on the density distribution of data points, identifying high-density regions as part of the cluster and others as noise, thus creating a hierarchical tree of clusters for different levels of granularity. There is no requirement to predefine the number of clusters to be created, and they are robust to noise. The DTW distance matrix is used for clustering. The outcome is regarded as favorable when the patterns of the signature in the clusters are similar, and are of substantial size.

For HDBSCAN, the most impactful parameters are maximum and minimum cluster size and minimum samples. We set a small minimum sample of 5, a larger minimum cluster size of 15, and the maximum cluster size to one-fourth of the total sample size to capture various patterns and maintain large clusters. The results of HDBSCAN on Niobrara marker signatures are shown in Figure 5.8. The template of each cluster is the barycenter of the samples in the cluster.

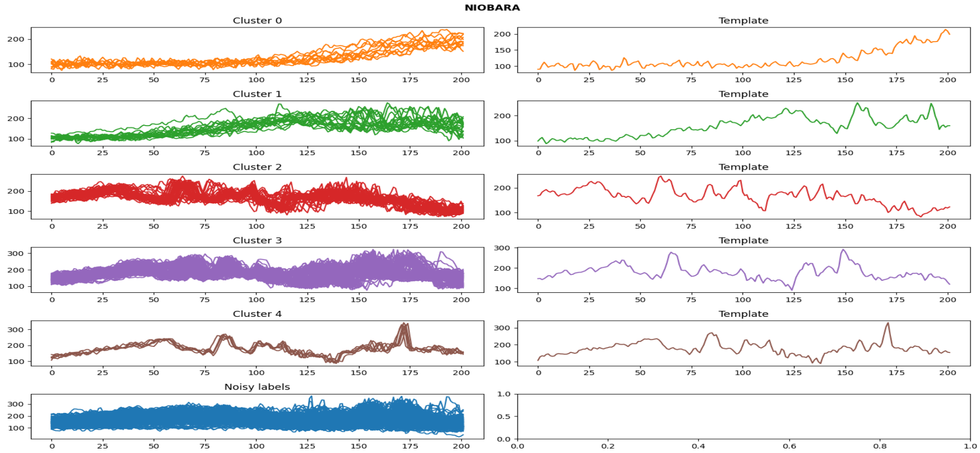


Figure 5.8: Results of the clustering algorithms on Niobrara signatures.

5.10 . Classification operation

This section outlines the DTW baseline, the new hybrid Deep Learning (DL) architectures, and the classification section of the framework. It discusses data preparation, DL model training mechanisms for optimal convergence in GeoTS, and post-processing to obtain predicted marker depths. Our objective is to determine the depth at which the formation starts. Marker signatures are used to identify this depth as discussed in Section 5.8.1.

5.10.1 . Baseline

A marker signature template needs to be selected for the DTW baseline approach. Then, compute the DTW distance between this signature template and all the gamma-ray signals extracted from the well using a sliding window approach. The template of the largest cluster obtained after clustering 5.9 is selected as the marker signature template. The smallest DTW distance refers to the depth of closest similarity between the signature template and the observed gamma ray. This depth is returned as the predicted marker depth. The time required for the Industrial baseline for multiple time sequences has a complexity of $O(n \cdot m \cdot s \cdot W)$ where n is the number of wells, m is the number of markers,s is the window size, and W is the average well length. Thus, we can propagate only one marker signature at a time. Figure 5.9 shows the DTW process where we see the signature template on the left side and the query sequence in which we need to find the best match on top. The similarity matrix indicates the region of minimum distance. Figure 5.10 shows the result of propagating all three marker templates on Well A. The dark blue indicates a low DTW distance, and the yellow indicates a high one. The disadvantage of DTW is that the method is based on pure statistics. It is heavily dependent on the signature template, making it rigid. This issue is overcome by using the

DL models, which allows us to include all variations of marker signatures to learn the template pattern.

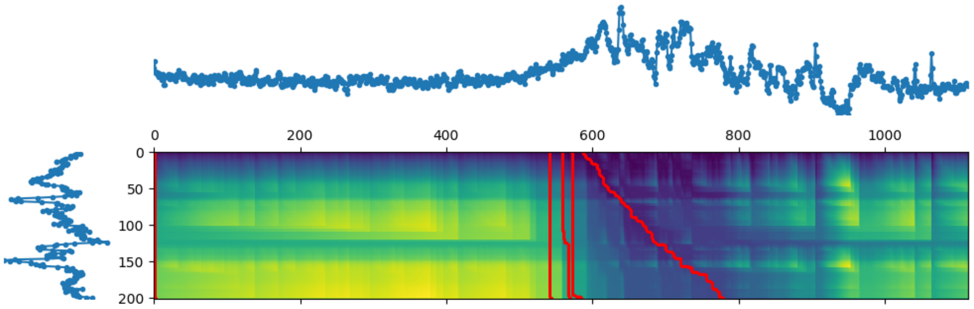


Figure 5.9: DTW match process for NIOBRARA.

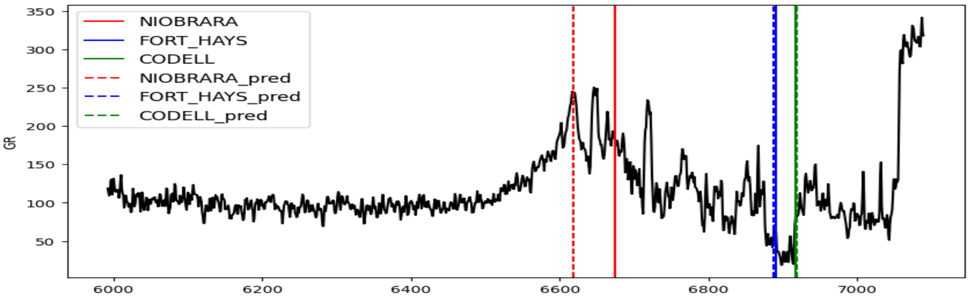


Figure 5.10: DTW result for propagation.

5.10.2 . Designed Hybrid models

It appears that hybrid models such as LSTM-FCN and XCM, which incorporate parallel networks of CNNs and RNNs, tend to exhibit improved performance compared to these models used individually or structured within a series network architecture. Taking inspiration from them, three new architectures were built using the LSTM, FCN, and parts of the XCM model as building blocks. These models are explained in this section.

LSTM-XCM is the augmentation of LSTM with XCM submodules. We combine an LSTM parallel network with the existing 1D and 2D parallel networks of the XCM. The results are concatenated and the second part follows the XCM architecture.

LSTM-FCN-2dCNN consists of a parallel network of LSTM, FCN, and 2DCNN layers. The output of the 2DCNN is reshaped to allow for the concatenation. In this case, after the concatenation of the outputs of the parallel network, the result is passed through the dropout and linear layer.

LSTM-2dCNN does not contain the FCN part of the LSTM-FCN-2DCNN model. This experiment was performed to understand the importance of the FCN part.

5.10.3 . Data preparation for DL model

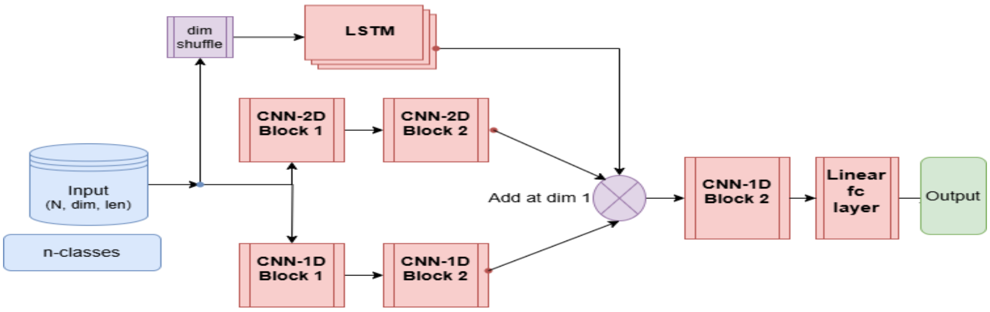


Figure 5.11: LSTM-XCN model architecture.

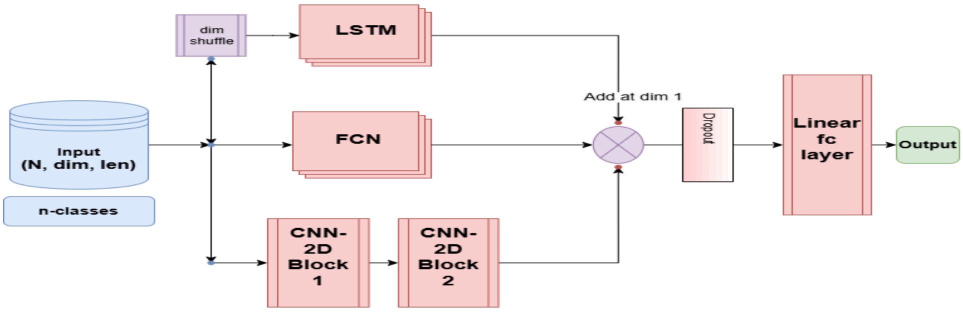


Figure 5.12: LSTM-FCN-2DCNN model architecture.

The results of the clustering operation help create a set of wells with valid marker signatures. We consider the largest cluster to be the one that represents, at best, the marker signature. Clusters having a similar template to the largest one are also considered as part of the learning dataset. From a dataset of 650 wells, 400 wells were filtered and used for training the DL model. The well-log marker signatures with their associated depths, latitude, and longitude are extracted. This is the input to the classification model. The classification model is trained to categorize sequences into four classes: 'None', 'Niobrara', 'Codell', and 'Fort Hays'. 'None' indicates depths without markers. As seen in the well logs in Section 5.8.1, most logs belong to the 'None' class, creating a class imbalance problem. To address the problem, a subset of the total samples is selected to create the final 'None' category.

5.10.4 . DL model training

Hyperparameter tuning enables DL models to achieve better convergence. Selecting a correct learning rate (Lr) influences whether the model's optimization will reach global minima and the time required for training the model. The Lr range test [367] is used to select Lr. Two training policies have been introduced to minimize fluctuations during regularization. The Lr is reduced if the validation loss increases over three epochs, enhancing the stability of the convergence. The second one stops the training if there is no significant improvement in validation loss for eight epochs. These policies reduce training time and help avoid overfitting or underfitting. This method brings more stability to the validation curve. The hyperparameter tuning is performed using the Tsai [301] and fastai [195] libraries. Cross-Entropy loss is applied for backpropagation using an Adam optimizer [230].

5.10.5 . Marker Prediction

The trained model is applied to the complete well-log data using a sliding window to generate probability curves to predict marker depths. A moving average filter is applied to smooth the probability curve. The predicted marker top depth is the one with the highest probability. Figure 5.13 illustrates the probability curve for well A and the extracted marker depth.

The time complexity of using the classification models after training is $O(n \cdot s \cdot W)$, where n is the number of wells, s is the sliding window length, and W is the average well length. This complexity is independent of the number of markers, as probability curves for all markers are generated simultaneously. Additionally, the approach does not depend on a single template like in the Industrial baseline, enhancing its robustness.

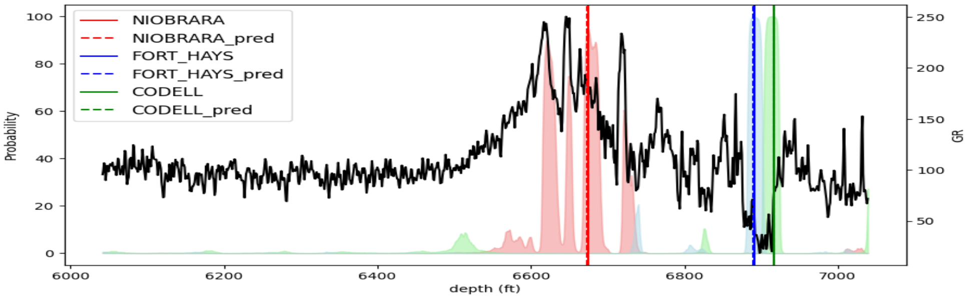


Figure 5.13: Propagation results for LSTM-XCM.

5.11 . Experiments and Results

This section presents the results of various classification models in GeoTS and discusses the experiments that identify key factors influencing these results. It also addresses the impact of different window sizes. The evaluation metrics from Section 5.8.2 were applied for model comparisons. The experiments were carried out over ten runs to account for randomness in the data selection and model weights initialization. The training dataset consists of 3,500 signal samples across four classes. When testing for prediction, the sample size increases significantly; for a well log of 7,000 ft, there are 7,000 test samples, leading to about one million samples across 150 wells.

When applying the classification models to predict the marker depth over the whole well, the recall varies over different models. Figure 5.14 shows the recall with a threshold of 10ft for the DL models and DTW baseline. Models like

LSTM, FCN, and Resnet have a low recall, with FCN performing the worst, followed by Resnet. LSTM-2dCNN and LSTM-XCM emerged as the best performers, exhibiting a high accuracy of around 95% and low variation, indicating robustness. The DTW baseline averages around 80%, similar to that of the recall of the inception model.

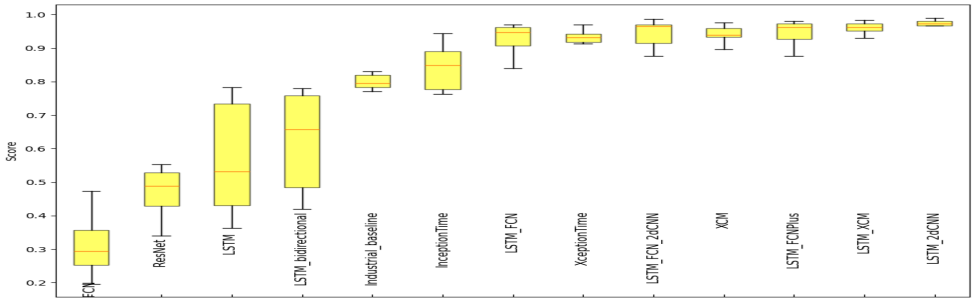


Figure 5.14: Recall score.

The test has been performed to check for model convergence by calculating the accuracy for a balanced testing dataset as discussed in Section 5.10.3. We get an accuracy greater than 90% for all the deep learning models in such a scenario. We conclude that since we decimate the ‘None’ class, some models have difficulty classifying the ‘None’.

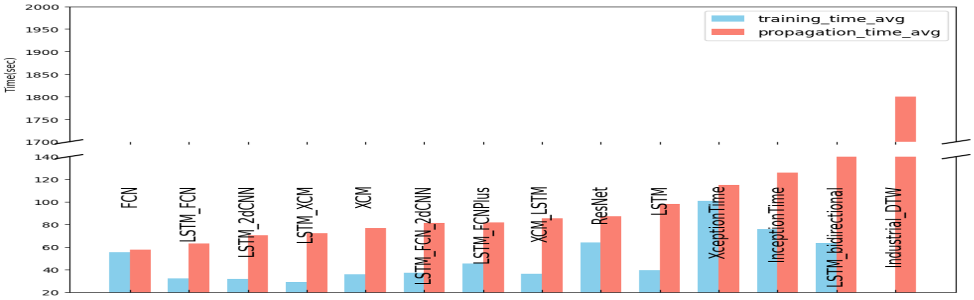


Figure 5.15: Time consumed in training and propagation.

Figure 5.15 compares training and prediction times, demonstrating the advantages of the proposed GeoTS framework, as it reduces the propagation time by more than 20 compared to the DTW approach when used to detect three markers. The process is also faster due to the use of GPU’s for the deep learning models. The experiments run on NVIDIA RTX 2000 Ada GPU. The time gap will increase as the number of markers to be detected increases.

5.11.1 . Detailed model performance analysis

To understand the prediction behavior of the models, we pick three models, one with low recall, one with average, and one with high recall: LSTM-bidirectional, LSTM-FCN, LSTM-XCM. The results for LSTM-bidirectional, LSTM-FCN, and LSTM-XCM on well B are presented in Figure 5.16. The probability curves computed are steep and less dispersed for models with higher recall.

The difference in accuracy and recall helps us understand that the results for a DL model on the real-world dataset can be quite different than when checked over a testing subset. This is particularly true when the data are noisy and can have minor variations over time or space. The LSTM-bidirectional model mispredicts the depths of the Niobrara, Codell, and Fort Hays markers, while the LSTM-FCN only mispredicts Codell. This does not imply entirely incorrect predictions, as actual depths also show high probability values. However, the result of the LSTM-XCM model shows high probability only close to actual depths, leading to correct predictions.

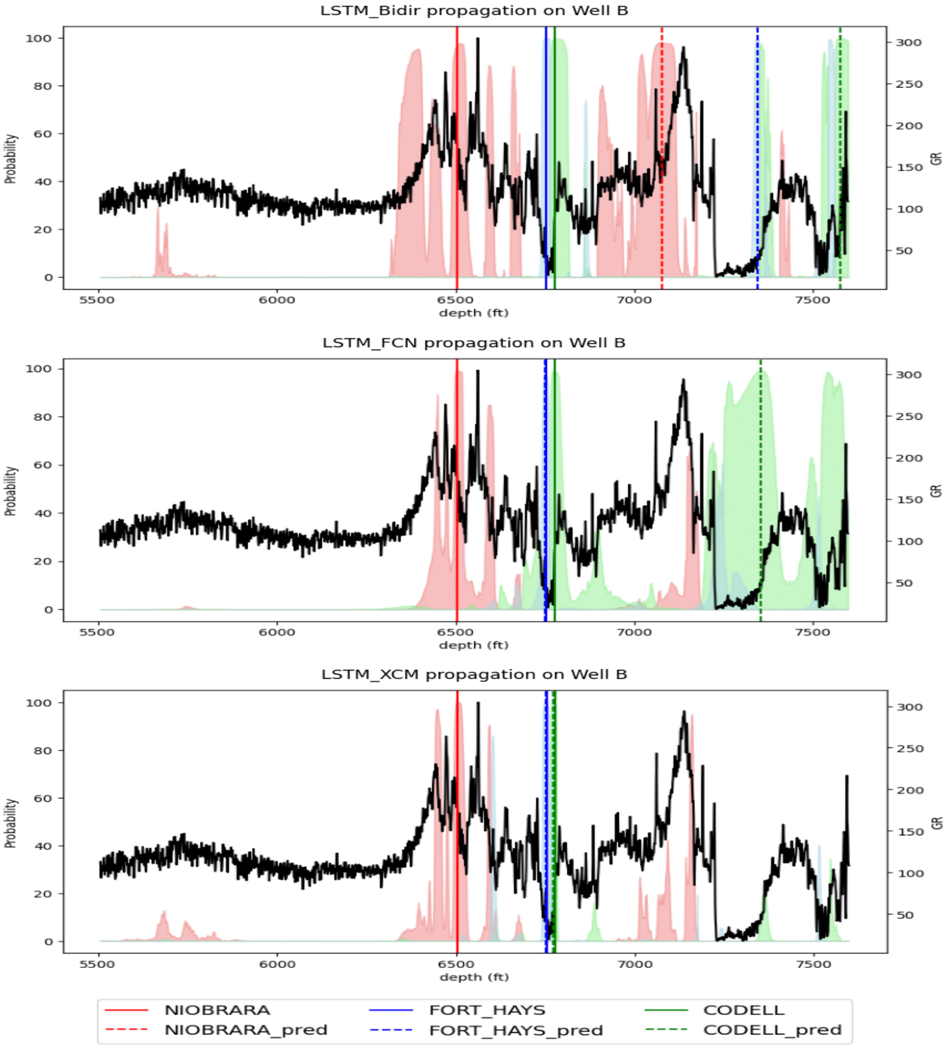


Figure 5.16: Propagation result on Well B.

The MAE is analyzed to compare model performance. LSTM-2dCNN, LSTM-XCM, and Xception have a low MAE as shown in Figure 5.17.

5.11.2 . Sliding Window size

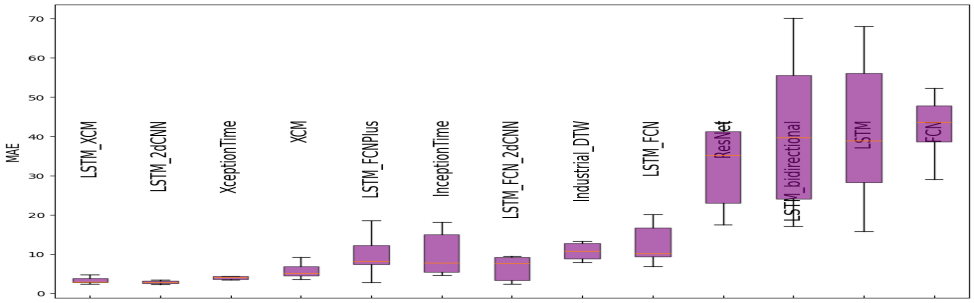


Figure 5.17: Mean Absolute Error.

The window size used to extract the signature should effectively represent the geological event of the marker; thus, it is an important parameter. A large window size can lead to clipped results at the well-log's start and end, which is especially problematic when markers are at these locations. Padding can mitigate this issue. Conversely, a short window size may not capture the geological event.

Methods	WS_201	WS_301	WS_101	WS_51
LSTM_FCN	0.97	0.92	0.84	0.84
LSTM_XCM	0.97	0.95	0.73	0.65
LSTM_2dCNN	0.97	0.98	0.93	0.66
LSTM_FCNPlus	0.96	0.93	0.65	0.66
InceptionTime	0.95	0.84	0.91	0.66
XCM	0.93	0.79	0.95	0.66
LSTM_FCN_2dCNN	0.94	0.96	0.76	0.62
XceptionTime	0.90	0.92	0.95	0.64
LSTM_bidirectional	0.76	0.83	0.72	0.59
LSTM	0.56	0.68	0.68	0.64
ResNet	0.53	0.66	0.39	0.76
FCN	0.27	0.41	0.79	0.85

Table 5.8: Results for different methods and window sizes (WS).

Table 5.8 illustrates the impact of various window sizes on the recall. The green cells highlight the optimal size. A window of 201ft has yielded good recall in most cases, and we used this size for all experiments. Smaller or larger window sizes tend to decrease final recall.

5.11.3 . Model Interpretability

We implement Grad-CAM [355], which can be applied to all models in the benchmark. Grad-CAM helps us with model interpretability and understand the model decision-making process at a deeper level. It is crucial in complex architecture, using a combination of different models, to gain insight into the

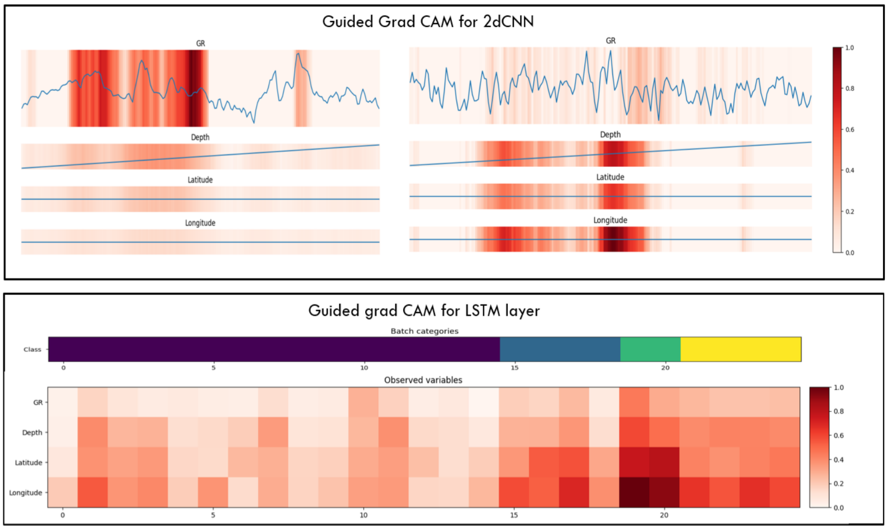


Figure 5.18: Grad-CAM implementation for the LSTM-2dCNN model.

success and failure of each part. Unreasonable predictions could have a reasonable explanation or vice versa. It allows us to tune models to make them more robust. It enables us to identify the distribution of feature importance in relation to the different classes. We can determine if the model has been able to pick the correct features to distinguish the classes.

We have adapted the Grad-CAM function for multivariate time series and propose a novel visualization. We implemented three variations, one for each RNN, 1dCNN, and 2dCNN layer, as they are standard building blocks for time series classification models. This would allow us to implement Grad-CAM on most time series models. To explain the functionality and effectiveness of the Grad-CAM function, we have performed a detailed analysis of the LSTM-2dCNN model.

Figure 5.18a represents the results of Grad-CAM for the 2dCNN part of the network. It can be seen that the "None" class has a higher dependency on the depth, latitude, and longitude compared to the GR signal, as opposed to the marker classes, where the dependency is higher on the signal. Figure 5.18b represents the results of Grad-CAM for the LSTM part of the network. The input to the LSTM is a multivariate time series with a single time step. Thus, we aggregated the results on the input variables of GR, depth, latitude, and longitude. The first subplot illustrates the signal categories, and the second the Grad-CAM results. It can be observed that the GR has a lower dependency for the 'None' class than for the marker classes. The 'None' class has a high dependence on the other variables is higher, while the reliance is lower for the different markers.

The Grad-CAM method helps us understand each parallel network's feature importance in such complicated hybrid models. It can be concluded that the parallel sub-model specializes in different input features with respect to the output classes, allowing the hybrid model to outperform the sub-models trained individually.

With respect to the specific execution of the experiments, we have conducted the deployment of GeoTS on the Wyoming oil fields, consisting of 5600 wells. Clustering according to the well location is performed using OPTICS [27]. For each geo-cluster, we select the most represented formations to be used for marker propagation using GeoTS. The process is conducted using a pre-existing Kubeflow pipeline. Table 2 presents the recall for the propagation of four formation sequences from the four different geo-clusters previously obtained. The sequences are Bg-W (Big George and Werner formations), Bgc-Bc-Fc (Big George coal, Badger coal, and Felix coal formations), Fh-La-Al (Fox Hill, Lance, and Almond formations), and LS-LM-LL (Tk unconformity, Lance shale, and Lance lower formations). We used a 200 ft window size for the processing. s We can clearly see that LSTM-XCM and LSTM-2dCNN perform better than the DTW algorithm. However, it can also be observed that the recall for the Fh-La-Al and LS-LM-LL sequences is lower overall and is closer to the recall of DTW. We have realized that the signatures of these markers do not follow a distinctive pattern in the GR well log. With no pattern being followed, even manually, we are unable to position the marker correctly using just the GR well log. This indicates the need for additional logs such as density, resistivity, and sonics, but these logs were not available in the use case. Some of the formations may have also been identified using seismic data. As for the formation sequences of Bg-W and Bgc-Bc-Fc, we have been able to increase the recall by more than 50%. The incorrect predictions were cases in which the GR log at the depth being similar to the marker signature. Figure 5.19 represents the case in which Big George coal has been inaccurately picked due to a similar pattern present at an incorrect depth. With this study, we can conclude that we need to consider not only the local features of the marker signature but also the global features of the well log between the markers.

Sequence	DTW	LSTM-FCN	LSTM-XCM	LSTM-2dCNN
Bg-W	0.21	0.71	0.73	0.74
Bgc-Bc-Fc	0.31	0.85	0.84	0.85
Fh-La-Al	0.25	0.45	0.55	0.48
LS-LM-LL	0.33	0.46	0.48	0.47

Table 5.9: Deployment results for Wyoming (Recall).

5.12 . Related Work

Knowledge Base (KB) construction is a recurrent problem in industry and research and includes problems of data extraction, cleaning, and integration [119].

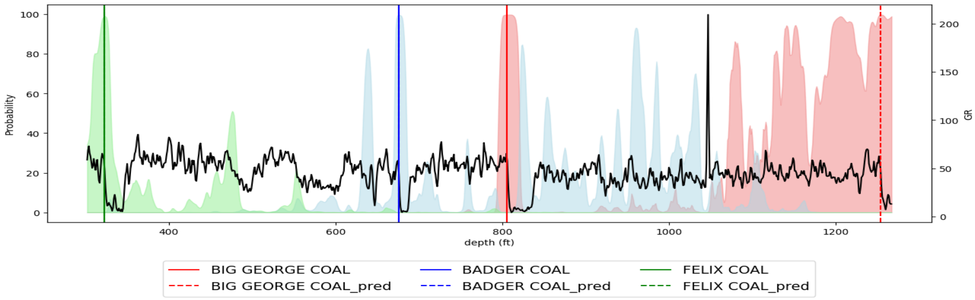


Figure 5.19: Wyoming propagation result.

A significant amount of work has been done in recent years on the automatic construction of knowledge bases. However, the first step of KB construction, which is defining a global schema with the aim of populating the KB, still requires manual effort [423]. Several previous studies were mainly focused on extracting data from unstructured data, such as texts. Open Information Extraction systems are not concerned with the integration of extracted entities and their properties from different sources with unified names. Because of this limitation, the resulting knowledge bases may represent the same entity multiple times with different names [423]. Other techniques, such as Biperpedia [161], use search engine query logs in addition to text to discover attributes. This process involves numerous trained classifiers and corresponding labeled training data. Most of the automatic KB construction systems were focused on retrieving facts and entities from unstructured datasets. To our knowledge, integrating the existing structured sources in the knowledge bases has not been considered in the process of constructing the KB automatically.

A large number of publications focused only on schema matching. In this context, schema matching identifies the correspondences between similar elements belonging to different schemas. IntelliLIGHT [159] is a system that looks in large-scale structured data sets, which aims to locate and retrieve needed data in a specific domain. It proposes a method that ranks the main data tables, taking as output the ones having a higher score. PROCLAIM is a very different approach to the problem; instead of ranking the best available schema among different data sources, it provides a unified standard schema from all sources and generates a global schema for a domain automatically. UFO [237] is a data structure expressing various representations of the same concept as a data object and is capable of recognizing and mapping such objects in different data sources automatically. The WebTable system [80] is a search engine that ranks tables scraped from the web. In this approach, AcsDB is introduced as a database that contains a corpus of statistics on schema elements that is used to compute the probability of an attribute (the number of schemas containing the attribute divided by the total number of schemas) and the probability of an attribute conditioned on another attribute. WebTable autocompletes a schema (suggests additional related attributes for a given set of attributes) by using the probability of pair attributes in different schemas to provide additional synonyms. In contrast, PROCLAIM focuses on all characteristics of all attributes to find similar attributes in the provided schemas. The main goal of PROCLAIM is to discover the most complete global schema over the existing schemas in a domain.

5.12.1 . Algorithms

The DTW algorithm computes the optimal correlation between two logs by minimizing the cumulative distance along a correlation path [283]. It is used for well correlation as it accommodates the time variations and has better non-linear behavior than other correlation techniques [185]. In a case study conducted on 22 wells of the Karatube Oilfield in Kazakhstan, a minimum spanning tree algorithm was used to find well-pairs, and later, DTW was performed to see the correlation between them [441].

Active research is being conducted to use machine learning to automate well correlation. One hundred wells from Kansas state oil and gas fields, USA, were used to train and test a 1D CNN Autoencoder [124]. Nearest wells are computed based on proximity. During training, the autoencoder tries to recognize a series of matching points from the nearest wells to match geological well tops. These methods strongly depend on the reference wells, with respect to which the other wells are correlated.

5.13 . Conclusions

Compared to the huge work on pairwise schema matching, research on holistic schema matching for more than two sources is still a challenge. PROCLAIM is an efficient and effective way for schema matching and provides a consistent domain-specific attribute schema. Experiments show that thanks to our approach, we can gather automatically more than 80% of the vocabulary related to a domain and populate the knowledge bases with corresponding attributes from heterogeneous sources. In future work, our approach can be extended for handling new attributes from new sources and for enriching the set of labels by adding similar words from different thesauri and dictionaries. Given the results in integrating data sources in the domain, we applied the results to analyze well log data and demonstrate the deep learning model's efficiency, ease, and accuracy in the pattern identification task. We present GeoTS, a framework that implements the workflow for the complete process: the preprocessing of data, the model implementation, and the post-processing. The Grad-CAM implemented can be applied to many deep-learning models. It will help understand the logic behind the decision-making of neural networks better for time series. The benchmark evaluation section clearly shows the Grad-CAM's ability to recognize the input feature's importance.

The main advantage of using GeoTS is that it is a completely autonomous process. We achieve an increase in recall and faster computational time when compared with the DTW baseline. Most of the existing frameworks used for marker propagation in wells are dependent on the neighboring wells log. As the model is trained on hundreds of wells, it is not dependent on a particular well, thus allowing the results to be resilient to the noise in the well logs. Our new models, the LSTM-2dCNN and the LSTM-XCM, outperform the state-of-the-art and baseline models.

The approach has been validated with the Colorado and Wyoming datasets, and additional tests for new datasets are scheduled by the company. To solve the issues we encountered during the deployment for Wyoming for the future, we will be trying a sequence-to-sequence transformer model. The whole well-log can be given as an input, allowing the model to understand the relative depth dependencies between the markers.

6 - Conclusion and Perspectives

This HDR manuscript has presented an overview of my research activities during the last 15 years. I presented different results, examining various methodologies and concrete application scenarios, also linked to industry use cases. Chapter 2 argued how data modeling can be useful in the NoSQL arena. Specifically, we have proposed a comprehensive methodology for the design of NoSQL databases, which relies on an aggregate-oriented view of application data.
Chapter 3 demonstrates the effectiveness of a novel application of Large Language Models (LLMs) for integrating heterogeneous data sources into a graph database. Through a comprehensive methodology that includes data modeling, extraction, and integration, supported by technologies such as Neo4j and GPT-3.5-Turbo, complex data processing tasks can potentially be streamlined. Although the data modeling choices have been centered around one specific dataset, several steps, such as those related to the modeling of entities as well as the decision of where to store attributes, can be expanded to other use-cases, especially in the context of an educational environment. The evaluation of both Named Entity Recognition and Data Resolution tasks illustrates the effectiveness and efficiency of LLMs in handling diverse data types. The project highlights the synergy between human expertise in data curation and AI's capabilities: opening avenues for more nuanced and scalable research databases.
Going further, Chapter 4 shows a typical example of the heterogeneity, detailing how smart cities could generate lots of data and be highly distributed geographically. In this review, we have explored data acquisition, data storage, data processing, and data governance management issues. We detailed how data analysis is a key enabler for smart city services, especially in energy conversion, showing that most of the challenges come from effective data processing.
Finally, Chapter 5 examined applications of data integration and the data preparation step for AI algorithms, also including time series. I also presented GeoTS, a framework that implements the workflow for the complete data analysis process: the preprocessing of data, the model implementation, and the post-processing.

I showed through my research how much, during the last years, data integration has been a continuously evolving challenge that has given me the opportunity to achieve interesting results detailed in the publications listed in Appendix B.

In the following paragraphs, I will conclude this manuscript by discussing new research questions that are already intriguing me and that will guide my future projects.

Data modeling - DataFrames integration During my research, I have deeply explored the data integration field; however, there is still an abundance of interesting open challenges to be solved. In the near future, we will formalize a metamodel for DataFrames integration. DataFrames are tabular data structures that do not strictly belong to a schema or a database. Data is usually stored in .csv files. The goal of the first research objective is to define a unified and flexible framework for integrating and processing DataFrames across platforms like Spark, R, and Pandas. Addressing the lack of structured logic in current DataFrame technologies, I would explore the development of a metamodel for representing relationships within distributed data sources [3], accommodating schema evolution, and enabling natural language querying. The project aims to ensure schema integrity through rule-based constraints, thus aligning the metamodel with user intent [72]. By tackling challenges in interoperability, consistency enforcement, and user-driven validation, the metamodel seeks to provide a scalable, intelligent, and usable framework for structured data representation and analysis [227, 305].

Data modeling - LLM and graphs Given the interesting results already achieved in the context of the Vrailexia research project, one of my future objectives is to continue exploring LLM to better identify and integrate entities in graph data models. With respect to this second objective, it will be essential to study and analyze the characteristics of LLM [249]. Nowadays, data integration is expected to be correct and efficient, and at the same time provide the possibility to execute effective query plans [12]. This context can be seen as a new challenge for aligning and solving entities: LLM still makes a lot of errors in aligning entities. In this regard, following the approach already developed in the Human in the loop methodology [417], I would like to explore how multiple prompting steps with human feedback can improve the query process. In the first preliminary phase, it can be interesting to push the feedback of the experts and construct a fine-tuning data integration and query answering approach. The solution will also potentially use probabilistic approaches to similarity and explore the inclusion of syntactic, thematic, and functional relationships into the conceptual schema. Moreover, since model fine-tuning is difficult due to the lack of available ground truth, it is worthwhile investigating generating a synthetic dataset using LLMs that are specifically tailored to the use case [388, 12].

Preparing data for Artificial Intelligence Given the results in [I1], we would like to continue exploring geological data and extend the datasets by integrating more data in the process. With this objective, we want to analyze not only well log data but also report data provided by the expert during the drilling process and demonstrate that with this unified dataset, the learning model accuracy will increase in the pattern identification task. With this objective, we clean and query the data to perform an initial analysis. We then prepare the data for machine learning tasks. Following this, we use generative AI techniques for imputing and predicting unavailable data. After the prediction, we integrate the results into our dataset. The final objective will be to use the enhanced dataset for further data analysis. While progress has been made in the field, further efforts are needed to develop data integration with efficiency, generalization, and a unified structure.

Enhancing and enriching time series analysis From a data processing point of view, another research aspect that we will start exploring in the geophysics domain will be the application of reinforcement learning from human feedback (RLHF).

Deep learning models specialized for these tasks are already available; it will therefore be important to perform benchmarks against these models and demonstrate the added value provided by the RLHF.

Due to the application framework of this project, the parameters to be predicted are quantities that are naturally subject to physical constraints. It will therefore be necessary to propose methods capable of integrating these constraints into the prediction models, in order not only to make these predictions realistic in the light of the intended application but also to potentially reduce the uncertainties associated with them.

Classical regression and classification models are often used for predictions of well logs, electrofacies, or depth of geological formations. Traditional statistical models such as Markov models[298] associated with neural networks are used to predict electrofacies[181], and random trees are used to reconstruct missing well logs[67].

More recently, deep learning is also used for well correlation [131]. Thanks to the success in the field of generative artificial intelligence, transformer-type architectures have been applied to time series. SAITS[128] captures temporal dependencies by introducing self-attention masked diagonal matrices, improving time series imputation. PatchTST [296] implements time series patches for encoding input layers. TIME-LLM [217] proposes to reuse language models for time series prediction. [311] confirmed that SAITS's performance in log reconstruction was better than currently used models.

Recently, we have seen the development of agentic artificial intelligence [359] that enables the creation of complex autonomous systems. This approach can be used to create a system specialized in petrophysical interpretation, where the system would adapt to the local conditions of the studies to be produced. Contextual information is essential in the field of geosciences to produce correct studies, as deeply discussed in the literature [311].

In this field, I want to explore the creation of complex autonomous systems specialized in petrophysical interpretation, where the system would adapt to the local characteristics of the studies.

To perform this data integration step, we can apply Retrieval-Augmented Generation (RAG) [246] techniques and automate the process by exploring agentic RAGs [363]: an autonomous mechanism that not only leverages RAG to augment context but actively identifies and assimilates the most relevant contextual cues [22]. This enhanced system would dynamically adjust analytical models based on evolving geological signals, thereby improving predictive accuracy and model adaptability in complex, heterogeneous geological environments.

As a final step, by creating an autonomous system, we should be able to use feedback from petrophysicists to improve the models. Human-based reinforcement learning (HLR) [101] appears to be appropriate for this task and constitutes a crucial step [435] to continuously refine not only the choice of contextual data but also the models themselves to ensure that the system is adapted to the geological constraints of the region under study. Methods similar to those proposed by [225] and [250] could be adapted to petrophysical interpretation. More specifically, RHLF could be explored in correcting the prediction of the top depth of formations crossing the wells, improving their alignment with the expert's geological intuition [51]. The project interested two companies, and a joint CIFRE PhD thesis project has been submitted to the ANR.

Smart cities and data integration Data analysis is a key enabler for smart city services and, vice versa, the smart city domain brings into the traditional data processing pipeline [438]. Nowadays, there is no meta-model accessible, which can draw the details about the smart city environment. In this context, it is essential to explore and develop a solution that integrates and makes possible standard operations that hide the heterogeneity of the components. The challenges behind the optimization reside in data, and a lot of interest is rising in the database research community [146, 394].

With respect to the specific aspect of applying machine learning and artificial intelligence algorithms, we are witnessing exponential growth and opportunities in the future. The main objective will be to use real data, as opposed to simulated, since our study tends to show overall greater performance and better results. Artificial intelligence is here to stay for energy conversion and the energy sector as a whole, and these techniques provide a myriad of opportunities to enhance the performance of traditional processes. In this context, we aim to explore the adoption and utilization of data integration techniques to enable algorithms to be applied to real data and produce insights in the energy field.

Global database for health The failure rate of drug development in oncology is extremely high (∼95%). There is a consensus in the oncology community that personalized therapies and precision oncology are the way to improve the success of new treatments in oncology. The goal of the REMISSION (*Rapid Evaluation of Molecular & Immune Status for Stratified Immunotherapies in ONcology*) program is to bring that effort to a next level of precision medicine and personalized oncology by implementing innovative techniques of fresh tissue explorations to better characterize the patient's disease biology and drug target expression. The main objective of my task I am managing is to design a database that is able to capture all relevant information about patients. Data will come from different sources: 1) eCRF ("Cahier d'observation électronique", a numeric booklet of data about the patient), where we can find the biological information of the patient in a specific format (either through .csv files, or through a database connection to the eCRF database). 2) data from Gustave Roussy or Clinical centres, where we will find specific information about the experiments, and that will be: raw data for cells (.fcs), exported data for cells (.csv), raw data for soluble factors (.txt), exported data for soluble factors (.csv). The integration of all these data sources will converge in the Portrait database, conceptualizing a global schema on top of Spark et Parquet storage layer [427, 60, 434, 199]. The global schema will abstract general mappings between the columns of the dataframes abstracted from the raw data in order to materialize for the analysis only the specific data that will be useful for the analysis and prediction algorithms. The main challenge will be not only in the abstraction of the matching between the columns of different sources [255] but also in the efficient extraction and integration of the instances given the enormous amount of data that we must handle (each clinical sample for a single patient will consist of more that 600 MB of data in average [220]).

Data for Physics Finally, I am setting up advanced/long-term collaborations on integrating physics data with CEA, Nairobi University, and multiple European partners. The aim of this new project is to analyze data coming from circular colliders that are in activity. My task is the development and deployment of a conceptual model integrating both simulation and experimental raw data. This model will allow machine learning and Artificial Intelligence applications to better conceive the next generation FCC (Future Circular Collider). I have already completed the first work on integrating, modeling, and aligning beam position measurements [I2]. The results are promising: my research demonstrated that these data can align in an effective manner with the prediction algorithms. Next steps will leverage key signal attributes such as betatron tune, amplitude, and noise-to-amplitude ratio to classify BPMs (Beam Position Monitors) into correct, suspected faulty, and faulty categories. The first results of the experiments under study show that the methodology successfully identifies already labeled problematic BPMs. We are working to improve the model by combining wavelet transform and Long Short-Term Memory (LSTM) networks. The wavelet transform provided a time-frequency decomposition of the signals, capturing transient features, while the LSTM model learned long-term temporal dependencies, further enhancing signal denoising capabilities.

In parallel, we are exploring how to evaluate HER data quality by identifying anomalous BPMs and comparing our detection outputs with SOMA's harmonic analysis (SOft MAtter dynamics with Delaunay-based Neighbours Search). The first results show that Gramian Angular Difference Field (GADF [428]) demonstrates the highest sensitivity in detecting BPM issues, suggesting its metrics may best correlate with known failure modes.

New insights could be defined thanks to better integration and exploration of these data sources. We are finalizing an ANR (French National Research Agency) project proposal to finance this research. These projects are also partly funded by French/Kenyan grants that I obtained very recently.

The final objective of the project will be to design a comprehensive framework for BPM signal processing, combining advanced machine learning and signal analysis techniques. The results will contribute to improved signal quality and fault detection, allowing reliable and efficient particle accelerator operations.

A - Curriculum Vitae

Francesca Bugiotti

Assistant professor at CentraleSupélec and member of the Large-scale Heterogeneous DAta and Knowledge (LaHDAK) team of LISN (Laboratoire Interdisciplinaire des Sciences du Numérique) of Paris-Saclay University.

Contacts

Poste Actuel
Maître de conférences, classe exceptionnelle
CentraleSupélec - LISN

Page personnelle
`www.bugiotti.it`

CentraleSupélec
3, rue Joliot-Curie
91192, Gif-sur-Yvette
France

LISN
1, rue Raimond Castaing
91192, Gif-sur-Yvette
France

E-mail
francesca.bugiotti@centralesupelec.fr

francesca.bugiotti@lisn.upsaclay.fr

Working Activity

March 2015 - present

Maître de conférences - CentraleSupélec.
Classe exceptionnelle from September 2023.
Research activity on Data Modeling, Data integration, and Artificial Intelligence.

November 2013 - February 2015

Post-Doc - Inria - Institut National de Recherche en Informatique et en Automatique.
Postdoctoral research, supervised by Ioana Manolescu[1], on efficient storage of large volumes of heterogeneous data in the cloud.

April 2012 - October 2013

Post-Doc - Università Roma Tre.
Postgraduate research activity, supervised by Paolo Atzeni, on model management in databases and No-SQL data stores integration.

April 2011 - July 2011

Researcher - Inria - Institut National de Recherche en Informatique et en Automatique.
Invited research stay, supervised by Ioana Manolescu, on indexing and storing semantic data through the Amazon Web Services (AWS) platform.

April 2011 - July 2011

Researcher - Consip[2]
I collaborated with Consip as a consultant on the quality of a data migration for the new information system of the Italian State General Accounting Department (Ragioneria Generale dello Stato Italiano).

[1] https ://pages.saclay.inria.fr/ioana.manolescu/

[2] Consip (Concessionaria Servizi Informativi Pubblici) s.p.a. is a public stock company owned by Italy's Ministry of the Economy

September 2008 - December 2009 **Researcher** ISA s.r.l.[3]
Part-time research activity on data mining applied to clinical data.

Education

Training

2025 Paris-Saclay University - “Fundamentals of Management” - 24h - December 2024 - April 2025

2024 *VSS - Formation E-Campus* - Université Paris-Saclay - “Violences Sexistes et Sexuelles” 29/11/2024

MOC SoSafe LISN - “Cybersecurity Awareness Training”

MOC - FUN Platform - “Dyslexic Students in My Lecture Hall : Understanding and Helping” 04/2024

2023 *Master Classes* - “Value and Knowledge Education” - 8h - 10/05/2023 - 12/10/2023

“Diversity Fresco” - 4h - 05/06/2023

2021 Training in the supervision of doctoral students - Université Paris-Saclay - 21h Captivate

Education

11/2008 – 04/2012 **Università Roma Tre - PhD in Computer Science - Computer Science and Automation department**

- Thesis :“A model-oriented approach to heterogeneity”.
- Advisor : Prof. Paolo Atzeni[4]

01/2008 – 03/2009 **Università Roma Tre - IBM - Formit** [5]
Post lauream degree in IT governance : development, management and monitoring (*Governo dei sistemi informativi : gestione, sviluppo e monitoraggio*).

10/2005 – 12/2007 **Università Roma Tre**
Master degree in Computer Engineering (“Laurea Specialistica in Ingegneria informatica”).

- Title of the thesis :“Tools and methodology for model management problems”.
- Advisor : prof. Paolo Atzeni
- Final grade : 110/110 lode (maximum honors)

and Finance that operates in behalf of the State.

[3] ISA s.r.l. http ://www.isa.it/, is an Italian enterprise that provides software for small and medium companies. It is focused on ERP services and business intelligence.

[4] `http://www.dia.uniroma3.it/~atzeni/`

[5] FORMIT is a Foundation that performs activities of scientific research, technical support, analysis and industrial, financial and socio-economic evaluation to sustain migration processes and integration of technological systems in every field of society.

10/2002 – 07/2005 **Università Roma Tre**

First level degree in Computer Engineering (“Laurea in Ingegneria informatica”).

- Thesis :“Datalog rules management for data and schema translation”.
- Advisor : Prof. Paolo Atzeni
- Final grade : 110/110 lode (maximum honors)

Awards

03/2009 “Accenture” Outstanding Engineering Graduate Award.

07/2007 Participant in “IBM EMEA Best Student Recognition Event”, Nice.

Teaching Activity

My teaching activities began in 2007, with a course on Java programming and algorithms, and have continued since. As of March 2015, the teaching hours are at least 192 hours/TD. My activities are divided between lectures, tutorials and practical work. The courses in which I have taught have targeted a very diverse audience, including students at L1, L2, L3, and M2 levels. In 2024/25, my teaching load is 338 HEqTD.

[**Summary of teaching activities**]

Class	Level	Year	Hours	Students
CentraleSupélec				
Big Data	C/TP/TD	M$_2$	25	80
Algorithms for distributed systems	C/TP/TD	M$_2$	24	20
Infrastructures modernes et cloud	C/TP/TD	M$_2$	24	40
Cloud computing et distributed programming	C/TP/TD	M$_1$	25	100
Software Engineering	C/TP/TD	M$_1$	15	100
Object Oriented Programming	TP/TD	M$_1$	15	25
Algorithms and data structures	TP/TD	L$_3$	25	60
Information systems and programming	TP/TD	L$_3$	15	35
Databases and Query Optimization	TP/TD	M$_1$	15	30
Hardware Architecture	TP/TD	M$_1$	15	25
Information systems	TP/TD	M$_1$	15	25
EXED - continuous education program/CentraleSupélec				
Relational databases and NoSQL	C/TP/TD	MS	7	30
Databases for Artificial Intelligence	C/TP/TD	MS	14	30
CentraleSupélec/Instutut Villebon Charpak				
Data Analysis	C/TP/TD	L$_3$	25	15
Essec/CentraleSupélec				
Big Data Algorithms, Techniques & Platforms	C/TP/TD	M2	25	120
CentraleSupélec/Erasmus Mundus				
Big Data Research Project	C/TP/TD	M2	25	20
TU-Berlin				
Data Analytics in Energy Sector Applications	C/TP/TD	M2	25	15
Advanced Database Design, Data Management & Integration	C/TP/TD	M2	25	15
Computer science and programming methods for Energy engineering	C/TP/TD	M1	25	15
Universitá Roma Tre				
Databases	TP/TD	L3	25	60
Object Oriented Analysis and Design	TP/TD	M2	25	15
IT Governance	C/TP/TD	M2	25	40
Java programming and Algorithms	TP/TD	L3	25	80
Remedial Mathematics	TP/TD	L2	25	30
Universitá della Tuscia				
Big Data	C/TP/TD	M2	25	15

M$_1$ = Master 1st Y, M$_2$ = Master 2nd Year, L$_3$ = Bachelor Year 3rd MS = Master for Professional employees - C = Cours, TD = Exercise classes, TP = Laboratories

Teaching Responsibilities

2021 - Present **BSc in Artificial Intelligence, Data & Management Sciences**

- Institutions : ESSEC/CentraleSupélec.
- Program academical director since September 2022.
- Scientific responsible of the bachelor.
- Leader of the development of the scientific program.
- Participation in the accreditation process and the interview with the CTI (Commission des Titres d'Ingénieur) and the CEFDG (Commission for the evaluation of management training and diplomas).

2022 **BSc HEPTA - Bachelor Hautes Etudes Pluridisciplinaires pour Top Athlètes**

- Institutions : ESSEC/CentraleSupélec/SciencePo/INSEP.
- Participation in the development of the scientific program.

2022 - Present **Comité Social d'Administration d'Etablissement (CSAE) CentraleSupélec**

Elected in December 2022 as a member of the Social Committee of Establishment Administration representant of the CFDT.

Research

Data is everywhere, comes from various sources, and can be represented and stored using heterogeneous structures. Integrating and combining data that are conceptually related but structurally different has historically posed challenges for driving research, innovation, and developing new solutions.

These challenges are the core of my research : my interests have always focused on the analysis and integration of heterogeneous databases.

Thanks to the availability of an evolving set of tools and technologies I have had access to since my master's studies, I have had the opportunity to explore and implement different data integration strategies (i) independent transformation of schema and data models, (ii) uniform access to NoSQL databases, (iii) management of data from the Semantic Web in cloud architectures, (iv) efficient integration of "big data" independent of the model, (v) using Large Language Model techniques to integrate graph databases, and in the most recent years, (vi) the application of Artificial Intelligence techniques on such integrated data.

The different use cases I explored confirmed that data integration is challenging and useful in a variety of situations, which include both industrial (i.e., two similar companies need to merge their databases or want to analyze data coming from different external sources) and scientific (i.e., combining research results from different laboratories or machines) domains.

A large part of my earlier work was concerned with the challenge of defining the characteristics of a global model for merging data in a general and full-comprehensive structure. This model also played a pivotal role in the dynamics of translating and potentially moving data from one schema to another (round-trip problem). This approach was useful for defining a more software-oriented data-integration set of techniques specifically dedicated to NoSQL data stores. I contributed to the definition of the architecture of the integration platform, the operations it exposes, and the query strategies it implements. I have been involved in defining the strategies for integrating the different database management systems into the platform. To explore the limits and opportunities of the emerging could-computing framework, I provided a solution for indexing RDF datasets using SimpleDB, a key-value store provided by AWS (Amazon Web Services).

In the task of integrating tabular data and information stored in pdf documents, I defined an approach that automatically retrieves information from them. The analysis step takes advantage of a pre-trained BERT model and applies two consecutive fine-tuning steps. My work focussed on the creation of a general data set that used techniques that identified lexicons and ran pattern recognition on documents. This was the first approach toward natural language analysis in my research after the early explorations in sentiment analysis. Enriching data on graph data models using the new Large Language Models (LLM) techniques for extracting entities and concepts from texts has been the next step of this research topic explored in a European Erasmus+ project. I contributed to the definition of a method that enriches a graph database using LLM, integrating multiple data sources in different formats and languages. The model captures complex relationships between entities that are not identifiable when considering each data source independently.

More recently, my research also benefitted from the fact that heterogeneous data is at the core of the Artificial Intelligence revolution. Data fuels machine learning models and shape the outputs of Artificial Intelligence software. In this context, I am exploring how much Artificial Intelligence algorithms present complex challenges for data integration to ensure that the results provided are trustworthy, and I am challenging the Data for Artificial Intelligence research topic in the context of an RHU project that aims to profile patients for personalized treatments in precision oncology. My main objective is to contribute to the definition, design, and implementation of the core database (PORTRAIT) that will integrate clinical data and experimental data (1GB for each patient) and provide them in the best format to AI run-time profiling procedures.

Lastly, I am setting up advanced/long-term collaborations on integrating physics data with CEA, Nairobi University, and multiple European partners. The aim of the project is to analyze data coming from circular colliders that are in activity. My task is the development and deployment of a conceptual model integrating both simulation and experimental raw data. This model will allow machine learning and Artificial Intelligence applications to better conceive the next generation FCC (Future Circular Collider). I have already completed the first work on integrating, modeling, and aligning beam position measurements. The results are promising : my research demonstrated that these data can align in an effective manner with the prediction algorithms. New insights could be defined thanks to better integration and exploration of these data sources, and we are finalizing an ANR (French National Research Agency) project proposal to finance this research. These projects also are partly funded by French/Kenya grants that I obtained very recently.

The industrial and real case studies that have enriched the theory's development opportunities are illustrated in more detail in the project section.

Supervision

PhD candidate co-supervision

October 2023 - present

Shwetha Salimat

- Title : GNN network - Exploration and application in the energy domain.
- Director : Frédéric Boulanger, (CentraleSupélec - LMF) 25% of the supervision
- Co-Supervisors : Francesca Bugiotti (CentraleSupélec - LISN) **50% of the supervision**, and Sylvain Wlodarczyk (SLB) 25% of the supervision.
- Publications and manuscripts : [I1], [I3], [S6], [I6]
- Funding : CIFRE with SLB.

October 2023 - Present

Quentin Bruant

- Title : Advanced techniques and artificial intelligence for the correction of linear and non-linear imperfections in the design of colliders of the future
- Director : Barbara Dalena (CEA) 40% of the supervision
- Co-supervisors : Valérie Goutard (CEA) 30% of the supervision, and Francesca Bugiotti (CentraleSupélec - LISN) **30% of the supervision**.
- Publications : [S2], [S3]
- Funding : CEA.

PhD Thesis co-supervision

2019-2023

Molood Arman

- Title : Weakly unsupervised approaches for building knowledge bases from geological and petrophysics heterogeneous data sources.
- Director : Nacéra Seghouani (CentraleSupélec - LISN) 30% of the supervision
- Co-supervisors : Francesca Bugiotti (CentraleSupélec - LISN) **40% of the supervision**, and Sylvain Wlodarczyk (Schlumberger) 30% of the supervision.
- Publications : [I8], [I11]
- Funding : CIFRE with SLB.
- Currently : Data Scientist - VOSSLOH [6]

Post-Doc supervision

March 2025

Charles NdungÚ Ndegwa

- Titolo : Data for AI : a new data model for predicting the best architecture for the next-generation colliders.
- Supervisor : Francesca Bugiotti

2017

Adnan El-Moussawi

- Title : Optimized storage of Graph data in Cloud Infrastructures.
- Funding : IT4BI
- Supervisors : Nacéra Seghouani, Francesca Bugiotti
- Publications : [I12]

[6] https ://www.vossloh.com/

Research Ingeneer Supervision

2025 - present

Jyotishka Das

- PhD candidate starting in October 2025
- Title : metadata modeling on medical data.
- Project : REMISSION RHU Project[7]
- Referees : Francesca Bugiotti - Paul-Henry Cournede (CentraleSupélec - MICS)

PhD Thesis Follow-up Committees[8]

Quentin Delamea

- Title : Modeling, proof and optimization of a fault-tolerant orchestrator (ArmoniK) on distributed and elastic architectures.
- Supervisors : Janna Burman, Stèphane Vialle, Jérôme Gurhem
- Referees : Francesca Bugiotti, Pierre Sutra

2022 - present

Hugo Gabrielidis

- Title : High-performance machine learning and data analytics for next-generation railway design.
- Supervisors : Stèphane Vialle, Filippo Gatti
- Referees : Francesca Bugiotti - Christian Cremona

Member of PhD Committee

November 29th 2024

Candidate : Tomasz Boczek

- Title : Developing IT architecture for electric vehicles charge point operators : Poland as the case study
- Director : Dr. Tetyana Morosuk (Faculty III, TU Berlin)
- Jury : Francesca Bugiotti (CentraleSupélec - LISN), Stefan Elbel (TU Berlin)

September 20th 2023

Candidate : Hiba Khalid

- Title : Detecting, Repairing, and Enhancing Raw Metadata
- Director : Dr. Esteban Zimanyi (Université libre de Bruxelles)
- Jury : Dimitris Sacharidis (Université Libre de Bruxelles), Mahmoud Sakr (Université Libre de Bruxelles), Darja Solodovnikova (University of Latvia), Francesca Bugiotti (CentraleSupélec - LISN)

October 20th 2020

Candidate : Amine Ghrab

- Title : Graph data warehousing
- Director : Dr. Oscar Romero Moral (Universitat Politècnica de Catalunya)
  Co-director : Dr. Esteban Zimanyi (Université libre de Bruxelles)
- Jury : Stijn Vansummeren (Université libre de Bruxelles), Hannes Voigt (Empreses d'Alemanya), Francesca Bugiotti (CentraleSupélec - LISN)

Master Thesis Supervision

2024/2025

Pavlo Poliuha : "Middleware for Online Exploration of Big Data."

[7] The project is a collaboration between a Hospital and a University. The research is focussed on data problems and fully described in the section "Projects and Research Studies"

[8] The PhD student Follow-up committee (CSI : Comité de Suivi Individuel du doctorant) is a Committee made compulsory by article 13 of the ministerial decree of 25 May 2016, which evaluates the conditions of the PhD student training and the progress of his research.

- Andrés Gomez : “Data analysis for anomaly detection and noise reduction in Turn by Turn BPMs signals.”

2023/2024

- Julien Ye : “Disability, helps, and success : analysis and evolution in the Paris-Saclay University context.”
- Abdellah Oumida : “The New Loop : RAG-enhanced LLM for Graph Data Integration.”

2021/2022

- Shwetha Salimat : “A Hybrid GNN approach for predicting node data for 3D meshes”
- Konstantino Mira : “Energy analysis techniques from literature a full and systematic classification”
- Ernesto Cernusa : “Understand the feelings of music track to Python”
- Lin Siying : “Graph Neural Networks analysis”

2020/2021

- Hem Bhatt : “Comparative DER Data Analysis for Architecture for Energy Consumption Optimization and Control”
- Antony Joseph : “A classification of BigData techniques applied in energy sector for the development of new research approaches”
- Akshay Tayde : “Evaluation of the IT methodology applied for energy sector and management systems”

2019/2020

- Shinji Kaneko : “The forecast and impact of day-ahead electricity price in Germany”
- Pallavi Katihalli-Manjegowda :“Effective data integration in smart cities for energy analysis”

2018/2019

- Moditha Hewasinghage : “Modeling Strategy for Storing Data in Distributed Heterogeneous NoSQL Databases”

2011/2012

- Daniele Calabresi : “Integration of Oracle NoSQL into a Platform for the Management of Non-Relational Data Stores”
- Tommaso D’Amora : “Integration of Amazon DynamoDB into a Platform for the Management of Non-Relational Data Stores”
- Marco De Leonardis : “Statistical Databases Management : an Approach Based on Translation Rules”
- Luca Rossi : “Heterogeneous Data Management on Innovative Database Systems”,

2009/2010

- Luca Tracuzzi : “Methodologies for Data Translation between Heterogeneous Data Models”
- Simone Folino : “Definition of Operators into a Model Management System”
- Marianna Ciminiello : “Object-Relational Mappings using MIDST”
- Stefano Mazzoni : “Object to Relational Mapping : a Metamodeling Approach”, avec Raimondo Tanariva
- Raimondo Tanariva : “Object to Relational Mapping : a Metamodeling Approach”, avec Stefano Mazzoni

2008/2009

- Fabrizio Celli : “Model Independent Data and Schema Translation : a Runtime Approach”
- Andrea Gozzi :“Import and Export of Schema and Data into a Model Management Platform”

Scientific responsibilities

2024 - present **Elected member of the Scientific Council of CentraleSupélec**

2022 - present **MADICS**

2023 - present **M4CAST**

2019 - present **Steering committee member of HUB AI CentraleSupélec - LISN**
I am part of the Steering committee of the HUB AI of CentraleSupélec and I am a correspondent between the HUB and the LISN research laboratory.
The AI Hub was launched in 2020, with the support of the general management and the CentraleSupélec Foundation. At the crossroads of teaching, research, and innovation, the purpose of the HUB is to spread AI "made in CS". The HUB wants to create an ecosystem of students, doctoral and post-doctoral students, researchers, professors, and companies through partnerships and actions around entrepreneurship.

2020 - present **Co-responsible for the organization of seminars for the LaHDAK team and the Data Science Department**
Collaborative activities for the organization of weekly (LaHDAK team) and monthly (Data Science Department) seminars.

2018 - present **Member of the PhD board "Engineering for Energy and Environment" - of Università della Tuscia**
The main objectives of the doctoral board are to plan the core of the strategic activities of the research doctorate and to verify the status of the planned activities. Together, we design the strategic development of the research, try to set a link between the industry and the university, develop international collaborations, and set informative actions. In this PhD board, I represent data science research axe.

Program Committees and Reviewer

Conference organzation

2025 Handiversite 2025 Conference - April 3, 2025 - Creativity for Inclusion, Paris-Saclay, general chair.

2023-2025 Data-driven Smart Cities (DASC) 2023, 2024, and 2025, ICDE Workshops.

2023 European Conference on Advances in Databases and Information Systems (ADBIS) 2023, workshop-track chair.

Handiversite 2023 Conference - April 20, 2023 - Creativity for Inclusion, Paris-Saclay, member of the program committee.

Member of the program committee

2025 Gestion de Donnèes - Principes, Technologies et Applications DBA (2025), scientific program committee.

2024 - present 27th International Conference on Discovery Science 2024 and 2025.

2023 Multi-Armed Bandits for Knowledge Discovery (MAB-KG) 2023, ICDM Workshops.

2022 Gestion de Donnèes - Principes, Technologies et Applications DBA (2022), member of the DEMO program committee.

2014 26th International Conference on Scientific and Statistical Database Management.

External reviewer

2014-2025 External reviewer for the International Conference on Extending Database Technology (EDBT) in 2012, for the Data & Knowledge Engineering (DKE) Journal in 2013, 2023-2025, for the Proceedings of Very Large Data Bases in 2014, and for the ACM SIGMOD Conference in 2014, Journal of Information Systems in 2022-2025, MENACIS in 2022, DS4EIW in 2023-2025.

Research Projects

2024-2029 **REMISSION RHU[9] Project**

- Academic Partners : Gustave Roussy, Paris-Saclay University, Inserm, Centre de recherche des Cordeliers, CentraleSupélec, Unicancer, Société française d'immunothérapie du cancer (FITC)
- Industrial Partners : HiFiBiO, PegaOne, ImCheck, et la biotech Veracyte
- Website
- Co-responsible of the WP3.4, WP4.6, and WP5.6.

The failure rate of drug development in oncology is extremely high (95%). There is a consensus in the oncology community that personalized therapies and precision oncology is the way to improve the success of new treatments in oncology. The goal of the REMISSION (*Rapid Evaluation of Molecular & Immune Status for Stratified Immunotherapies in ONcology*) program is to bring that effort to a next level of precision medicine and personalized oncology by implementing innovative techniques of fresh tissue explorations to better characterize the patient's disease biology and drug target expression. The main objective of my task is to design a database that is able to capture all relevant information about the biological information of patients.

[9] Recherche Hospitalo-Universitaire en santé (RHU) - Research in Hospital and University collaboration focussed on Health problems

2020-2023 **VRAILEXIA European Erasmus+ Project**

- Academic Partners : Università della Tuscia, Università degli studi di Perugia, Panteion, University of Paris Nanterre, UCCL, CentraleSupélec, Universidad de Cordoba
- Industrial Partners : Tucep, Giunti, AEVA.
- Website
- *Scientific responsible for CentraleSupélec - Budget 30000 euros*

Le projet VRAILEXIA[10] is a European Erasmus+ project, prized by Unesco, which aims to change perception and develop a tool to overcome the main difficulties of dyslexics by strengthening their motivation and self-esteem. The project's main objective is to develop a digital platform based on AI to support dyslexic students. My role was the integration of data that comes from several tests, in several languages, for the evaluation of the profile of dyslexia and the effects of the use of the platform on the psychological aspects [S5], [J2], [I5], [J3], [N1].

2023-2026 **GEOTS**

- Partners : SLB, CentraleSupélec, LISN
- *Funding : CIFRE*
- Scientific Participant

In geoscience, it is necessary to study the lithography of the Earth's subsurface, which consists of different stratified layers called geological formations. This study performs well correlation task to model and characterize reservoirs. This operation links the beginning of specific geological formations called tops using measurements from drilled wells. Although data are abundant, the traditional algorithms used for well correlation are semi-automated, requiring significant time and high computational cost. We aim to introduce GeoTS, a Python library to apply cutting-edge time series classification models to perform well correlation in a completely automated setting. As input, it takes the drilling trajectory depth and gamma-ray well logs, which measure the natural radioactivity across the well depth trajectory. The top depths of the formations are predicted as an output. The gamma-ray signatures are extracted around the top depths assigned by geologists. Preprocessing is performed to clean and cluster these signatures using Dynamic Time Wrapping (DTW) distance and HDBSCAN. Implementation of existing deep learning architectures (FCN, InceptionTime, XceptionTime, XCM, LSTM-FCN) and new architectures (LSTM-2dCNN, LSTM-XCM) are performed.

2022 - present **BMP[11] trajectory analyses**

- Partners : CEA, University of Nairobi, LISN, CentraleSupélec
- *Scientific responsible of the collaboration for LISN/CentraleSupélec - Budget 60000 euros*

This research project has as its main objective to integrate and analyze data coming from Circular Colliders in collaboration with the CEA (Commissariat à l'énergie atomique et aux énergies alternatives[12]) and the University of Nairobi. After the discovery of the Higgs boson at the Large Hadron Collider (LHC), the particle physics community is exploring and proposing the next accelerators to address the remaining open questions. One of the studied possibilities is FCC (Future Circular Collider), a 100-km-long collider at CERN. In the project context, we are collecting data in different European countries. The focus is on the development and deployment of a conceptual model integrating both simulation and experimental raw data in order to run machine learning and artificial intelligence applications. A post-doc financed by the collaboration will join the LISN Laboratory in January 2025. I am already co-supervising a PhD thesis financed by CEA. An ANR project is under submission.

[10] Virtual Reality and Artificial Intelligence for Dyslexia

[11] Beam Position Monitors (BPM)

[12] https ://www.cea.fr/

2019 - present **IT4Energy**

- Partners : TU Berlin, LISN
- *Funding : TU Berlin*
- Scientific Participant

Energy transformation, often referred to as energy conversion, and green hydrogen technologies and efficiencies are critical components of the plan to achieve net-zero CO_2 emissions. In this research collaboration we explore data in this area. We aim to study how the use of artificial intelligence (AI) and machine learning (ML) tools could built opportunities to accelerate and optimize the performance and efficiencies of energy conversion tasks [J4].

2020-2023 **PROCLAIM**

- Partners : SLB, CentraleSupélec, LISN
- *Funding : CIFRE*
- Scientific Participant

The objective of this project is to explore and define information extraction approaches and to build learning models to obtain a knowledge base from documents drilling (cuttings), laboratory reports of core data analysis and geological studies in order to automatically provide the a priori information necessary to interpreting logs using the available knowledge and the business rules [I8], [I11].

2018-2021 **B-GRAP**

- Partners : CentraleSupélec, LISN
- *Funding : CentraleSupélec*
- Scientific Participant

The definition of effective strategies for graph partitioning is a major challenge in distributed environments since an effective graph partitioning allows to considerably improve the performance of large graph data analytics computations. In this project we studied and defined a multi-objective and scalable Balanced GRAph Partitioning (B-GRAP) algorithm to produce balanced graph partitions. B-GRAP is based on Label Propagation (LP) approach and defines different objective functions to deal with either vertex or edge balance constraints while considering edge direction in graphs. The experiments are performed on various graphs while varying the number of partitions [J5], [I12].

2018-2020 **SATT [13] DataForYou**

- Partners : SATT Paris-Saclay, CentraleSupélec, LISN
- *Funding : SATT*
- Scientific Participant

Participation in the SATT DataForYou project aimed at supporting the creation of the start-up DataForYou, which aims to build tools to support local authorities (for example, town halls, departmental administrations) in France. The project's objective was to integrate data for optimizing services provided to citizens by relying on behavioral analysis tools. In this project, I was involved in coaching an engineer on the data integration batch.

[13] SATT Paris-Saclay is the Technology Transfer Accelerator Office of the Paris-Saclay Cluster.

2016-2017 **APIQA**

- Partners : LRI
- *Funding : LRI*
- Co-responsible of the project

This project had as objective to define a methodology that provides complete answers to queries over data accessible via Web APIs. The project focused on Twitter graph data for the beginning. The project defined a query engine that integrates real-time (or online) queries over the Twitter API with a local (or offline) data source. It was possible to build and maintain the data using a NoSQL graph datastore. Moreover, we focused on different ways of organizing data on the offline datastore, in order to improve the performance of queries and the completeness of the results [N2], [I14], [I15].

2013-2017 **NOAM and ONDM**

- Partners : Università Roma Tre
- *Funding : Università Roma Tre*
- Scientific Participant

NoSQL Abstract Model (NOAM) is a logical approach to the NoSQL database design problem [N6] and aims at exploiting the commonalities of various NoSQL systems. It is based on an intermediate, abstract data model where aggregates are units of distribution (to support scalability) and consistency (to the extent it is needed). Some intermediate representations can be implemented in target NoSQL datastores, considering their specific features and providing effective support for scalability, consistency, and performance [J6], [I17]. ONDM (Object-NoSQL Datastore Mapper), is the framework [N5] that supports NOAM approach. It provides application developers with a uniform programming interface, as well as the ability to map application data to different data representations and can be used, in an effective way, for performing the experiments during the design of a NoSQL database [N6].

2013-2016 **ESTOCADA**

- Partners : INRIA, UC Sant Diego
- *This work has been partially funded by the Datalyse "Investissement d'Avenir" project, by the associated INRIA-Sillicon Valley OakSad team, and the KIC ICT Labs Europa activity*
- Scientific Participant

A novel system capable of exploiting side-by-side a practically unbound variety of DMSs, all the while guaranteeing the soundness and completeness of the store, and striving to extract the best performance out of the various DMSs. Our system leverages recent advances in the area of query rewriting under constraints, which we use to capture the various data models and describe the fragments each DMS stores [D1], [N3], [I16], [N4].

2013 **GENDATA (Università Roma Tre - Politecnico di Milano)**

- Partners : Politecnico di Milano, Università di Bergamo, Università di Milano, Politecnico di Torino, Università di Bologna, Università La sapienza, Università Roma Tre, Università di Salerno, Università della Calabria
- *Funding : PRIN (Italian National Project)*
- Scientific Participant

The work regarding data models continues within the GENDATA European project `http://gendata.weebly.com/` that aims at building the abstractions, models, and protocols for supporting a network of genomic data, making them available for genome servers located in the major biologist laboratories in the world. I started to collaborate to the project within the working packages involving Università Roma Tre investigating about the model design, the query language and the model standardization.

2011-2012 **SOS**

- Partners : Università Roma Tre
- *Funding : Università Roma Tre*
- Scientific Participant

Save Our Systems (SOS) is a common programming interface [D10] to NoSQL systems. Its goal is to support application development by hiding the specific details of the various systems. I contributed to the definition to the architecture of the platform, the operations it exposes and the query strategies it implements. I have been involved in defining the strategies for integrating the NoSQL data stores into the system. I also participated in the definition of the data storage techniques that are used in each datastore in order to perform operations the interface exposes [J9], [I19].

2011 **AMADA**

- Partners : INRIA
- *Funding : This work has been partially funded by the KIC EIT ICT Labs activity "Clouds for Connected Cities" 2011 and an AWS in Education research grant*
- Scientific Participant

During my internship I contributed to the AMADA project : a platform [N7], [D2] for storing Web data (XML documents and RDF graphs) based on the Amazon Web Services (AWS) cloud infrastructure. I provided a solution for the problem of indexing RDF datasets by using SimpleDB, a key-value store provided by AWS. I contributed to the definition and development of four indexing strategies [I16], [B1].

2009-2012 **MISM**

- Partners : Università Roma Tre
- *Funding : Università Roma Tre*
- Scientific Participant

Model Independent Schema Management (MISM) is a platform for model management that offers a set of operators to manipulate schemas. I designed and implemented the algorithm that gives one solution to the round-trip engineering problem considering the main model management operators (merge, diff, and modelgen) implemented according to model-independent and model-aware approaches based on MIDST supermodel [J11], [N9].

2008-2013 **MATRIX - EXL**

- Partners : Università Roma Tre, Central Bank of Italy
- *Funding : Central Bank of Italy*
- Scientific Participant

I collaborated with the Bank of Italy, supporting the implementation of EXLEngine, a tool that manipulates statistical data at high level in terms of entities of statistical models such as time series. We proposed (*i*) a language, EXL, has been defined for the declarative specification of statistical programs, (*ii*) an approach for the translation of EXL code into executables in various target systems has been developed, and (*iii*) a concrete implementation, EXLEngine. The approach leverages schema mappings as an intermediate specification step, in order to facilitate the translation from EXL towards several target systems [J7], [I18].

2008-2012 **MIDST-RT**

- Partners : Università Roma Tre
- *Funding : Università Roma Tre*
- Scientific Participant

Model-Independent Schema and Data Translation-RunTime (MIDST-RT) is a platform based on MIDST but that implements a runtime approach. I contributed to the definition, design, and implementation of the MIDST-RT algorithm, given the schema of the source database and the model of the target one generates views on the operational system that expose the underlying data according to the corresponding schema in the target model. The implemented approach generates views automatically, based on the Datalog rules for schema translation [J10], [I21], [N8], [N10].

2005-2012 **MIDST**

- Partners : Università Roma Tre
- *Funding : Università Roma Tre*
- Scientific Participant

Model-Independent Schema and Data Translation (MIDST) is a platform for model-independent schema and data translation based on a meta-level approach over a wide range of data models (Relational, OR, OO, ER, XML). I contributed to the extension of MIDST *supermodel* (a general model handled by the platform that describes the various data models in terms of a small set of *basic constructs*) and I also implemented some core software components like the Datalog-SQL translation engine giving some ideas about the evolution of the platform [N10].

Industry Collaborations and Contracts

2021 - present **Transvalor**

Participation to a research-contract for the development of a chaire de recherche [I6], [I7], [I9], [I10]. Research topic : data integration.

2022 - present **Tissium**

Consultant for CentraleSupélec for the definition of a shared database storing all data coming from the experiment and the research. Definition of an internship that will be supervised during the next summer.

2018 - present **Vires - Msc Software**

Co-supervision of projects developed with students of CentraleSupélec.

2017 - present **SLB**

Co-supervision of multiple projects, focusing on data analysis and integration, developed with students of CentraleSupélec. Co-Supervision of two PhD thesis.

Research dissemination

2024 **Olympiades des Sciences de l'Ingénieur**

2020-2021 **Mentorat Solinum**
Mentoring of Solinum company for data analysis and the application of AI algorithms. Action in collaboration with the school's Entrepreneurship cell. The objective of this collaboration was to provide an environment of scientific expertise to consolidate this project born during the COVID crisis. In my action I participated in the improvement of Soliguide, a tool from the company Solinum, which makes it possible to guide social action thanks to Artificial Intelligence.

2018-2023 **"A volte ritornano"**
Series of scientific seminars held by ex-students of the Liceo Paolo Ruffini in Viterbo. The aim of the seminars is to make research accessible to all and to communicate to people the passion for science. For high school students, the initiative aims to provide useful tools in the choice of their future career, and to inform on current lines of scientific research.

2021 **Project "Summer school for young female students"**
The University of Paris-Saclay aims to introduce students to disciplines traditionally neglected by girls, in a vast recruitment pool that includes priority areas. I participated in two weeks of initiatives for high school and middle school students through a seminar : "Big Data : the incredible opportunities for storage and interpretation" and the participation in several newsgroups.

2018-2025 **Dalkia : Women's Energy In Transition** [14]
Member of the jury - *Dalkia challenge* aims to promote the place of women in the field of energy, to highlight rich and exemplary careers and to encourage young women to join these professions, by participating in its promotion among students. I contributed to this project taking part in the evaluation of the candidates files and the jury.

Seminars and Publications

Seminars

March 7th 2025 Seminar - Université de Tours - Symposium - Blois, "Graph data representations and graph data models in different use cases".

May 30th 2024 Seminar - MADICS - Symposium - Blois, "Named Entity Recognition using Deep Neural Networks and Large Language Models".

October 24th 2023 Round Table - "SUN - Semaines des Usages du Numérique", - Paris-Saclay, Title : "Innovation and Handicap".

October 16th, 2023 Seminar - National Dyslexic Day - Paris, Title : "Digital and Artificial Intelligence Integrated Tools to Support Higher Education Students with Dyslexia ".

December 12th 2022 Seminar - CEA - In the art - DataIA - Paris-Saclay, Title : "Weakly supervised Named Entity Recognition using Deep Neural Networks".

[14] Dalkia is a subsidiary of EDF Group and a leading provider of energy services, with operations across France and further afield. Women's Energy In Transition award rewards and financially supports female students and professionals in activity, to encourage women to join training courses or professions related to the energy transition.

July 4th 2021 Seminar - CentraleSupélec - Paris-Saclay, Title : “The cartography of the Artificial Intelligence Research in CentraleSupélec”.

July 9th 2018 Seminar - TU-Berlin - Berlin - research group DIMA / DFKI, “Interpreting Reputation through Frequent Named Entities in Twitter”.

December 5th 2017 Seminar - TU-Berlin - Berlin - research team DIMA / DFKI, “Modeling Methodology for a uniform access to NoSQL systems”.

December 4th 2017 Invited tutorial - TU-Berlin - Berlin - research team DIMA / DFKI, Title : “Database Design for NoSQL Systems”.

April 22nd 2016 Seminar - Roma Tre - Rome - Database research group, “Flexible Stores and Data”.

Glossary of Scientific and Industrial Collaborations

LISN Laboratory

LaHDAK (Large-scale Heterogeneous DAta and Knowledge)

- Collaborators : *Nacéra Seghouani, Benoit Groz, Silviu Maniu, Gianluca Quercini*
- LaHDAK Website

Team A&O (Learning and Optimization)

- Collaborators : *Mathieu Kowalski*
- A&O Website

Team M3 (Models, Methods, and Multilingualism)

- Collaborators : *Bonneau Hélène*
- M3 Website

National (France/Italy)

Ecole Nationale Supérieure d'Informatique pour l'Industrie et l'Entreprise (ENSIIE)

- Collaborators : *Stefania Dumbrava*
- `https://web4.ensiie.fr/`

Paris Nanterre University

- Collaborators : *Quinio Bernard*
- `https://university.parisnanterre.fr//`

Università Roma Tre

- Collaborators : *Paolo Atzeni, Luca Cabibbo, Riccardo Torlone*
- `https://ingegneriacivileinformaticatecnologieaeronautiche.uniroma3.it/en/`

Politecnico di Milano

- Collaborators : *Stefano Ceri*
- `https://www.polimi.it/`

Liceo Scientifico Paolo Ruffini

- Collaborators : *Alessandro Ercoli*
- `https://www.liceopaoloruffiniviterbo.it/`

CNR (Consiglio Nazionale delle Ricerche)

- Collaborators : *Umberto Straccia*
- `www.cnr.it/people/umberto.straccia`

Università della Tuscia

- Collaborators : *Giuseppe Calabró*
- `https://www.unitus.it/`

INRIA (Institut national de recherche en sciences et technologies du numérique)

- Collaborators : *Ioana Manolescu*
- `https://inria.fr/fr`

CNRS (Centre national de la recherche scientifique) and Museum National d'Histoire Naturelle

- Collaborators : *Christophe Lavelle, Olivier Getcher*
- `https://biophysique.mnhn.fr`, `https://gechter.org/blog/`

SLB

- SLB is a global technology company. As of 2022, it is both the world's largest oilfield services company and the world's leader in digital solutions for subsurface and surface engineering.
- Collaborators : *Sylvain Wlodarczyk, Souaib Oizineb*
- `https://www.slb.com/`

Genvia

- A joint venture between CEA, SLB, Occitanie, Vinci, and Vicat to enable the H2 economy through the decarbonisation of industry.
- *Collaborators : Alessia Longobardi, Yan Hermann*
- `https://genvia.com/`

Dalkia

- Dalkia is a subsidiary of EDF Group and a leading provider of energy services, with operations across France and further afield.
- *Collaborators : Sylvie Jéhanno*
- `https://www.dalkia.com/careers/women-energy-in-transition/`

Institut Gustave Roussy

- Institut Gustave Roussy is a cancer research hospital ranked as the first leading cancer hospital in Europe and in the top five best-specialized hospitals in the world.
- Collaborators : *Aurélien Marabelle*
- `https://www.gustaveroussy.fr`

Generali Nantes

- Generali is an insurance company that offers its customers a complete range of insurance solutions (health, personal protection, assistance, property, and liability), savings, and asset management.
- Collaborators : *Michael Reis*
- `https://www.generali.fr`

CEA (Commissariat à l'énergie atomique et aux énergies alternatives)

- The CEA is a major research organization working in low-carbon energy (nuclear and renewable), digital technology, technology for medicine of the future, defense, and national security.
- Collaborators : *Barbara Dalena, Valérie Gautard, Adnad Ghirbi*
- `https://www.cea.fr`

Transvalor

- Transvalor is a French company that offers a unique solution platform able to simulate the overall manufacturing process, from the raw material to the product in-use properties.
- Collaborators : *Jose Alves, Patricia Renaud*
- `https://www.transvalor.com/`

Tissium

- Tissium is a company founded in 2013 whose objective is to disrupt the field of surgery and positively impact the lives of patients through the development of our platform of biomorphic programmable polymers.
- Collaborators : *Maria Pereira, João Maia*
- `https://tissium.com/`

Solinum

- Solinum is committed to developing and distributing innovative digital tools, ensuring universal access to information on social assistance and services.
- Collaborators : *Victoria Mandefield*
- `https://www.solinum.org/`

Vires - MSC Software - Exagon

- Hexagon is a leading global provider of information technology solutions. The company supports the development, testing, and validation of driver-assisted and fully autonomous driving technology.
- Collaborators : *David Mear*
- `https://hexagon.com/`

Central Bank of Italy

- The Bank of Italy is the central bank of the Republic of Italy. It is a public-law institution regulated by national and European legislation.
- Collaborators : *Luigi Bellomarini*
- `https://www.bancaditalia.it/`

Consip

- Consip (Concessionaria Servizi Informativi Pubblici) s.p.a. is a public stock company owned by Italy's Ministry of the Economy and Finance that operates in behalf of the State.
- Collaborators : *Gianna Caralla*
- `https://www.consip.it/`

ISA

- ISA s.r.l. is an Italian enterprise that provides software for small and medium companies. It is focused on ERP services and business intelligence
- Collaborators : *Giuseppe Materni*
- `http://www.isa.it/`

International

TU Berlin, Germany

- Collaborators : *Tatiana Morosuk, Ralf-Detlef Kutsche*
- `https://www.tu.berlin/`

University of Oulu, Finland

- Collaborators : *Ekaterina Gilman*
- `https://www.oulu.fi/`

Norwegian University of Science and Technology, Norway

- Collaborators : *Xiang Su*
- `https://www.ntnu.edu/`

UC Sant Diego - USA

- Collaborators : *Alin Deutsch*
- `https://ucsd.edu/`

Nairobi University - Kenya

- Collaborators : *Ian Kaniu, Kenneth Amiga Kaduki*
- `https://www.uonbi.ac.ke/`

B - Publications

Book Chapters (B)

B1 F. Bugiotti, J. Camacho-Rodríguez, F. Goasdoué, Z. Kaoudi, I. Manolescu et S. Zampetakis, « SPARQL Query Processing in the Cloud, » in *Linked Data Management.* CRC Press, 2014, p. 165-192.

Articles in international journals (J)

J1 E. Gilman, F. Bugiotti, A. Khalid et al., « Addressing Data Challenges to Drive the Transformation of Smart Cities, » *ACM Transactions on Intelligent Systems and Technology - [Scimagojr Q1]- To Appear - 63 pages*, 2024.

J2 A. Remadi, K. E. Hage, Y. Hobeika et F. Bugiotti, « To prompt or not to prompt : Navigating the use of large language models for integrating and modeling heterogeneous data, » *Data & Knowledge Engineering - [Scimagojr Q2]*, t. 152, n° 102313, 2024.

J3 A. Harfouche, B. Quinio et F. Bugiotti, « Human-Centric AI to Mitigate AI Biases : The Advent of Augmented Intelligence, » *Journal of Global Information Management - [Scimagojr Q2]*, t. 31, n° 5, p. 1-23, 2023.

J4 K. Mira, F. Bugiotti et T. Morosuk, « Artificial Intelligence and Machine Learning in Energy Conversion and Management, » *Energies - [Scimagojr Q2]*, t. 16, n° 23, 2023, issn : 1996-1073. 🔗 doi : `10.3390/en16237773`.

J5 A. E. Moussawi, N. B. Seghouani et F. Bugiotti, « BGRAP : Balanced GRAph Partitioning Algorithm for Large Graphs, » *Journal of Data Intelligence - [Scimagojr Q1]*, t. 2, n° 2, p. 116-135, 2021.

J6 P. Atzeni, F. Bugiotti, L. Cabibbo et R. Torlone, « Data modeling in the NoSQL world, » *Computer Standards & Interfaces - [Scimagojr Q2]*, t. 67, 2020.

J7 P. Atzeni, L. Bellomarini, F. Bugiotti et M. D. Leonardis, « Executable Schema Mappings for Statistical Data Processing., » *Distributed Parallel Databases - [Scimagojr Q2]*, t. 36, n° 2, p. 265-300, 2018.

J8 N. B. Seghouani, F. Bugiotti, M. Hewasinghage, S. Isaj et G. Quercini, « A Frequent Named Entities-Based Approach for Interpreting Reputation in Twitter, » *Data Science and Engineering - [Scimagojr Q2]*, t. 3, n° 2, p. 86-100, 2018.

J9 P. Atzeni, F. Bugiotti et L. Rossi, « Uniform access to NoSQL systems, » *Information Systems - [Scimagojr Q1]*, t. 43, p. 117-133, 2014.

J10 P. Atzeni, L. Bellomarini, F. Bugiotti, F. Celli et G. Gianforme, « A runtime approach to model-generic translation of schema and data., » *Information Systems - [Scimagojr Q1]*, t. 37, n° 3, p. 269-287, 2012.

J11 P. Atzeni, L. Bellomarini, F. Bugiotti et G. Gianforme, « MISM : A Platform for Model-Independent Solutions to Model Management Problems, » *Journal of Data Semantics - [Scimagojr Q2]*, t. 14, p. 133-161, 2009.

Full articles in international conferences and workshops (I)

I1 S. Salimath, S. Wlodarczyk et F. Bugiotti, « GeoX : Explainable neural network for time series classification, a geoscience case study, » in *KDD 2025 - Applied Data Science Track, - [Core Rank A*]*, to appear, t. 2, 2026, p. 12.

I2 Q. Bruant, B. Dalena, F. Bugiotti et al., « Emittance Tuning of the FCC-EE high energy booster ring, » in *European Physical Society Conference on High Energy Physics*, 2025.

I3 S. Salimath, S. Wlodarczyk et F. Bugiotti, « Responsible AI : Training deep learning model efficiently, » in *New Trends in Database and Information Systems - ADBIS - [Core Rank C]*, to appear, 2025.

I4 D. Sechet, F. Bugiotti, M. Kowalski, E. d'Hérouville et F. Langiewicz, « A Hierarchical Deep Learning Approach for Minority Instrument Detection, » in *International Conference on Digital Audio Effects (DAF) - [Core Rank B]*, t. to Appear, 2024.

I5 K. E. Hage, A. Remadi, Y. Hobeika, R. Ma, V. Hong et F. Bugiotti, « A multi-source graph database to showcase a recommender system for dyslexic students, » in *IEEE International Conference on Big Data, BigData 2023 - [Core Rank B]*, IEEE, 2023, p. 3134-3138.

I6 S. Salimath, F. Bugiotti et F. Magoulès, « A Hybrid GNN Approach for Predicting Node Data for 3D Meshes, » in *New Trends in Database and Information Systems - ADBIS - [Core Rank C] Short Papers*, sér. Communications in Computer and Information Science, t. 1850, Springer, 2023, p. 130-139.

I7 S. Li, J. Alves, F. Bugiotti et F. Magoulès, « A Comparison Study of Graph Neural Network and Support Vector Machine, » in *Distributed Computing and Applications for Business Engineering and Science (DCABES) - [Scimagojr - ranking in progress]*, 2022.

I8 R. G. Londono, S. Wlodarczyk, M. Arman, F. Bugiotti et N. B. Seghouani, « Weakly supervised Named Entity Recognition for Carbon Storage using Deep Neural Networks, » in *International Conference on Discovery Science (DS) - [Core Rank B]*, 2022.

I9 V. S. Meduri, J. Alves, F. Bugiotti et F. Magoulès, « Point-Cloud-based Deep Learning Models for Finite Element Analysis, » in *Distributed Computing and Applications for Business Engineering and Science (DCABES) - [Scimagojr - ranking in progress]*, 2022.

I10 V. S. Meduri, F. Bugiotti et F. Magoulès, « Point-Cloud-based Deep Learning Models for Finite Element Analysis, » in *Distributed Computing and Applications for Business Engineering and Science (DCABES) - [Scimagojr - ranking in progress]*, 2022.

I11 M. Arman, S. Wlodarczyk, N. B. Seghouani et F. Bugiotti, « PROCLAIM : An Unsupervised Approach to Discover Domain-Specific Attribute Matchings from Heterogeneous Sources, » in *International Conference on Advanced Information Systems Engineering (CAiSE) - [Core Rank A]*, t. 386, Springer, 2020, p. 14-28.

I12 A. E. Moussawi, N. B. Seghouani et F. Bugiotti, « A Graph Partitioning Algorithm for Edge or Vertex Balance, » in *Database and Expert Systems Applications DEXA - [Core Rank C]*, sér. Lecture Notes in Computer Science, t. 12391, Springer, 2020, p. 23-37.

I13 M. Hewasinghage, N. B. Seghouani et F. Bugiotti, « Modeling Strategies for Storing Data in Distributed Heterogeneous NoSQL Databases, » in *International Conference on Conceptual Modeling (ER) - [Core Rank A]*, sér. Lecture Notes in Computer Science, t. 11157, Springer, 2018, p. 488-496.

I14 N. Bennacer, F. Bugiotti, J. Galicia, M. Patricio et G. Quercini, « Eliminating Incorrect Cross-Language Links in Wikipedia, » in *Web Information Systems Engineering (WISE) - [Core Rank B]*, 2017, p. 109-116.

I15 N. Bennacer, F. Bugiotti, M. Hewasinghage, S. Isaj et G. Quercini, « Interpreting Reputation Through Frequent Named Entities in Twitter, » in *Web Information Systems Engineering (WISE) - [Core Rank B]*, 2017, p. 49-56.

I16 F. Bugiotti, D. Bursztyn, A. Deutsch, I. Ileana et I. Manolescu, « Invisible Glue : Scalable Self-Tunning Multi-Stores, » in *Conference on Innovative Data Systems Research (CIDR - [Core Rank A]*, 2015.

I17 F. Bugiotti, L. Cabibbo, P. Atzeni et R. Torlone, « Database Design for NoSQL Systems., » in *International Conference on Conceptual Modeling (ER) - [Core Rank A]*, 2014, p. 1-7.

I18 P. Atzeni, L. Bellomarini et F. Bugiotti, « EXLEngine : executable schema mappings for statistical data processing, » in *International Conference on Extending Database Technology (EDBT) - [Core Rank A]*, 2013, p. 672-682.

I19 P. Atzeni, F. Bugiotti et L. Rossi, « Uniform Access to Non-relational Database Systems : The SOS Platform, » in *International Conference on Advanced Information Systems Engineering (CAiSE) - [Core Rank A]*, 2012, p. 160-174.

I20 F. Bugiotti, F. Goasdoué, Z. Kaoudi et I. Manolescu, « RDF Data Management in the Amazon Cloud, » in *Workshop on Data analytics in the Cloud (DanaC)*, 2012.

I21 P. Atzeni, L. Bellomarini, F. Bugiotti et G. Gianforme, « A runtime approach to model-independent schema and data translation, » in *International Conference on Extending Database Technology (EDBT) - [Core Rank A]*, 2009, p. 275-286.

Demonstrations in international conferences (D)

D1 F. Bugiotti, D. Bursztyn, A. Deutsch, I. Manolescu et S. Zampetakis, « Flexible hybrid stores : Constraint-based rewriting to the rescue, » in *International Conference on Data Engineering (ICDE) - [Core Rank A]*, 2016, p. 1394-1397.

D2 A. Aranda-Andújar, F. Bugiotti, J. Camacho-Rodríguez et al., « AMADA : Web Data Repositories in the Amazon Cloud, » in *International Conference on Information and Knowledge Management (ACM CIKM) - [Core Rank A]*, 2012.

D3 P. Atzeni, F. Bugiotti et L. Rossi, « SOS (Save Our Systems) : a uniform programming interface. for non-relational systems, » in *International Conference on Extending Database Technology (EDBT) - [Core Rank A]*, 2012, p. 582-585.

Articles and demos in national database conferences (N)

N1 Q. Bernard, A. Harfouche et F. Bugiotti, *Human-centric AI to mitigate AI biases : The advent of augmented intelligence*, AIM, éd., Conférence de l'Association Information et Management (AIM), 2022.

N2 N. B. Seghouani, F. Bugiotti, J. Galicia, M. Patricio et G. Quercini, *Élimination des liens inter-langues erronés dans Wikipédia*, M. Lebbah, C. Largeron et H. Azzag, éd., Conférence francophone sur l'Extraction et la Gestion des Connaissances (EGC), 2018.

N3 R. B. AL-Otaibi, F. Bugiotti, D. Bursztyn, A. Deutsch, I. Manolescu et S. Zampetakis, *Estocada : Stockage Hybride et Ré-écriture sous Contraintes d'Intégrité.* Journées des Bases de Données Avancées (BDA), 2016.

N4 F. Bugiotti, D. Bursztyn, A. Deutsch, I. Ileana et I. Manolescu, *Toward Scalable Hybrid Stores*, Italian Symposium on Advanced Database Systems (SEBD), 2015.

N5 F. Bugiotti, L. Cabibbo, P. Atzeni et R. Torlone, *How I Learned to Stop Worrying and Love NoSQL Databases*, Italian Symposium on Advanced Database Systems (SEBD), 2015.

N6 F. Bugiotti et L. Cabibbo, *A Comparison of Data Models and APIs of NoSQL Datastores.* Italian Symposium on Advanced Database Systems (SEBD), 2013.

N7 A. Aranda-Andújar, F. Bugiotti, J. Camacho-Rodríguez et Z. Kaoudi, *AMADA : Web Data Repositories in the Amazon Cloud.* Journées des Bases de Données Avancées (BDA), 2012.

N8 P. Atzeni, L. Bellomarini, F. Bugiotti et G. Gianforme, *A runtime approach to model-independent schema and data translation*, Italian Symposium on Advanced Database Systems (SEBD), 2009.

N9 P. Atzeni, L. Bellomarini, F. Bugiotti et G. Gianforme, *A platform for model-independent solutions to model management problems*, Italian Symposium on Advanced Database Systems (SEBD), 2008.

N10 P. Atzeni, L. Bellomarini, F. Bugiotti et G. Gianforme, *From Schema and Model Translation to a Model Management System*, British National Conference on Databases (BNCOD), 2008.

Submitted Papers (S)

S1 Q. Bruant, F. Bugiotti, B. Dalena et al., « AI techniques to improve optics measurements based on the Turn-by-turn Beam Position Monitors, » 2025.

S2 Q. Bruant, F. Bugiotti et B. Dalena, « Echo State Network analysis for Dynamic Aperture prediction, » 2024.

S3 Q. Bruant, A. Guelfane, Y. Zuo et al., « Anomaly detection and noise reduction in Turn by Turn BPMs signals of SuperKEKB main rings, » 2024.

S4 V. de Chaisemartin, F. Bugiotti, C. Lavelle et O. Getcher, « Correlation between health and the environment for foods, » 2024.

S5 E. Galliére, B. Lorenzi et F. Bugiotti, « Graph-Based Analysis of Dyslexic Profiles in University Using Unsupervised Clustering., » 2024.

S6 G. L. Houssaini, Y. Wang, Y. Hou, S. Salimath et F. Bugiotti, « Learning Models in the Context of Predicting Geological Markers Formation for Oil & Gas Drilling Processes, » 2024.

S7 S. Lemaachi, O. Chater, A. Longobardi, F. Bugiotti, Y. Herrmann et S. Wlodarczyk, « Failure Prediction in Electrolyzers with Interpretable Image-Based Deep Learning and Unsupervised Domain Adaptation, » 2024.

Bibliography

[1] Serge Abiteboul, Richard Hull, and Victor Vianu. *Foundations of Databases*. Addison-Wesley, 1995.

[2] Tanzina Afrin and Nita Yodo. A long short-term memory-based correlated traffic data prediction framework. *Knowledge-Based Systems*, 237:107755, 2022.

[3] Waqas Ahmed, Leticia Gómez, Alejandro Vaisman, and Esteban Zimanyi. Reconciling tuple and attribute timestamping for temporal data warehouses. *Proc. VLDB Endow.*, 34, 12 2024.

[4] AirNow. AirNow.gov.

[5] Anirudh Ajith, Chris Pan, Mengzhou Xia, Ameet Deshpande, and Karthik Narasimhan. InstructEval: Systematic Evaluation of Instruction Selection Methods. *arXiv preprint*, 2023.

[6] Mustafa A. Al Ibrahim. Uncertainty in automated well-log correlation using stochastic dynamic time warping. *Petrophysics - The SPWLA Journal*, 63(06):748–761, 12 2022.

[7] Eiman Al Nuaimi, Hind Al Neyadi, Nader Mohamed, and Jameela Al-Jaroodi. Applications of big data to smart cities. *Journal of Internet Services and Applications*, 6(1):25, Dec 2015.

[8] Ioannis Alagiannis, Renata Borovica-Gajic, Miguel Branco, Stratos Idreos, and Anastasia Ailamaki. NoDB: efficient query execution on raw data files. *Commun. ACM*, 58(12):112–121, 2015.

[9] Vito Albino, Umberto Berardi, and Rosa Maria Dangelico. Smart cities: Definitions, dimensions, performance, and initiatives. *Journal of Urban Technology*, 22(1):3–21, 2015.

[10] Tomé Albuquerque, Ricardo Cruz, and Jaime Cardoso. Ordinal losses for classification of cervical cancer risk. *PeerJ Computer Science*, 7:e457, 04 2021.

[11] Muhammad Intizar Ali, Feng Gao, and Alessandra Mileo. Citybench: A configurable benchmark to evaluate rsp engines using smart city datasets. In *International Semantic Web Conference (ISWC)*, pages 374–389. W3C, 2015.

[12] Syed Juned Ali, Iris Reinhartz-Berger, and Dominik Bork. How are llms used for conceptual modeling? an exploratory study on interaction behavior and user perception. In Wolfgang Maass, Hyoil Han, Hasan Yasar, and Nick Multari, editors, *International Conference on Conceptual Modeling (ER)*, pages 257–275. Springer Nature Switzerland, 2025.

[13] Ahmed Alnuaim, Ziheng Sun, and Didarul Islam. Ai for improving ozone forecasting. In *Artificial Intelligence in Earth Science*, pages 247–269. Elsevier, 2023.

[14] Gustavo Alonso, Natassa Ailamaki, Sailesh Krishnamurthy, Sam Madden, Swami Sivasubramanian, and Raghu Ramakrishnan. Future of database system architectures. In *International Conference on Management of Data (SIGMOD)*, page 261–262. ACM, 2023.

[15] Ali A Alwan, Azlin Nordin, Mogahed Alzeber, and Abedallah Zaid Abualkishik. A survey of schema matching research using database schemas and instances. *International Journal of Advanced Computer Science and Applications (IJACSA)*, 8(10), 2017.

[16] Rami Aly, Andreas Vlachos, and Ryan McDonald. Leveraging type descriptions for zero-shot named entity recognition and classification. In *International Joint Conference on Natural Language Processing (IJCNLP)*, pages 1516–1528. ACL, August 2021.

[17] Amazon. Amazon Mechanical Turk.

[18] Amazon Web Services. DynamoDB. http://aws.amazon.com/dynamodb. accessed February 2016.

[19] NG Nageswari Amma and F Ramesh Dhanaseelan. Privacy preserving data mining classifier for smart city applications. In *International Conference on Communication and Electronics Systems (ICCES)*, pages 645–648. IEEE, 2018.

[20] Amsterdam. Data portal Amsterdam.

[21] Mike Ananny and Strohecker Carol. Textales: Creating interactive forums with urban publics. In Marcus Foth, editor, *Handbook of Research on Urban Informatics: The Practice and Promise of the Real-Time City*, pages 68–86. IGI Global, Boston, 2009.

[22] R Anantha, T Bethi, D Vodianik, and S Chappidi. Context tuning for retrieval augmented generation. *arXiv preprint arXiv:2312.05708*, 2023.

[23] Mark A. Andersen. Defining log interpretation.

[24] Margarita Angelidou. The role of smart city characteristics in the plans of fifteen cities. *Journal of Urban Technology*, 24(4):3–28, 2017.

[25] R. Angles and C. Gutierrez. Survey of graph database models. *ACM Computing Surveys*, 40(1):1–39, 2008.

[26] Renzo Angles and Claudio Gutierrez. *An Introduction to Graph Data Management*. Data-Centric Systems and Applications. Springer, 2018.

[27] Mihael Ankerst, Markus M. Breunig, Hans-Peter Kriegel, and Jörg Sander. Optics: ordering points to identify the clustering structure. In *International Conference on Management of Data (SIGMOD)*, page 49–60. ACM, 1999.

[28] Apache. Apache activemq.

[29] Apache. Apache cassandra.

[30] Apache. Apache flink.

[31] Apache. Apache flume.

[32] Apache. Apache giraph.

[33] Apache. Apache hadoop.

[34] Apache. Apache hbase.

[35] Apache. Apache kafka.

[36] Apache. Apache nifi.

[37] Apache. Apache ozone.

[38] Apache. Apache Sedona, a cluster computing system for processing large-scale spatial data.

[39] Apache. Apache spark™ - Unified Analytics Engine for Big Data.

[40] Apache. Apache Storm.

[41] Apache. Apache tez.

[42] Apache. Apache Zookeeper.

[43] Isuri Anuradha Nanomi Arachchige, Le Ha, Ruslan Mitkov, and Johannes-Dieter Steinert. Enhancing named entity recognition for holocaust testimonies through pseudo labelling and transformer-based models. In *International Workshop on Historical Document Imaging and Processing (HIP)*, page 85–90. ACM, 2023.

[44] Patricia Arocena, Boris Glavic, Radu Ciucanu, and Renée Miller. The ibench integration metadata generator. *Proc. VLDB Endow.*, 9, 11 2015.

[45] Simran Arora, Brandon Yang, Sabri Eyuboglu, Avanika Narayan, Andrew Hojel, Immanuel Trummer, and Christopher Ré. Language models enable simple systems for generating structured views of heterogeneous data lakes. *Proc. VLDB Endow.*, 17(2):92–105, oct 2023.

[46] Paolo Atzeni, Francesca Bugiotti, Luca Cabibbo, and Riccardo Torlone. Data modeling in the nosql world. *Comput. Stand. Interfaces*, 67, 2020.

[47] Paolo Atzeni, Francesca Bugiotti, and Luca Rossi. Uniform access to NoSQL systems. *Inf. Syst.*, 43:117–133, 2014.

[48] Paolo Atzeni, Christian S. Jensen, Giorgio Orsi, Sudha Ram, Letizia Tanca, and Riccardo Torlone. The relational model is dead, SQL is dead, and I don't feel so good myself. *SIGMOD Record*, 42(2):64–68, 2013.

[49] Muhammad Babar, Fahim Arif, Mian Ahmad Jan, Zhiyuan Tan, and Fazlullah Khan. Urban data management system: Towards big data analytics for internet of things based smart urban environment using customized hadoop. *Future Generation Computer Systems*, 96:398–409, 2019.

[50] Antonio Badia and Daniel Lemire. A call to arms: revisiting database design. *SIGMOD Record*, 40(3):61–69, 2011.

[51] Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

[52] Jason Baker et al. Megastore: Providing scalable, highly available storage for interactive services. In *Conference on Innovative Data Systems Research (CIDR)*, pages 223–234, 2011.

[53] Angels Balaguer, Vinamra Benara, Renato Luiz de Freitas Cunha, Roberto de M. Estevão Filho, Todd Hendry, Daniel Holstein, Jennifer Marsman, Nick Mecklenburg, Sara Malvar, Leonardo Nunes, Rafael Padilha, Morris Sharp, Bruno Silva, Swati Sharma, Vijay Aski, and Ranveer Chandra. RAG vs Fine-tuning: Pipelines, Tradeoffs, and a Case Study on Agriculture. *arXiv preprint*, 2024.

[54] Daniela Ballari, M. Wachowicz, and Miguel Ángel Manso Callejo. Metadata behind the interoperability of wireless sensor network. *Sensors (Basel, Switzerland)*, 9:3635–51, 05 2009.

[55] S. K. Bansal. Towards a semantic extract-transform-load (etl) framework for big data integration. In *2014 IEEE International Congress on Big Data*, pages 522–529, June 2014.

[56] Marcello Barbella and Genoveffa Tortora. A semi-automatic data integration process of heterogeneous databases. *Pattern Recognition Letters*, 166(C):134–142, feb 2023.

[57] Sarah Barns. Smart cities and urban data platforms: Designing interfaces for smart governance. *City, Culture and Society*, 12:5 – 12, 2018. Innovation and identity in next generation smart cities.

[58] Carlo Batini, Stefano Ceri, and Shamkant B. Navathe. *Conceptual Database Design: An Entity-Relationship Approach*. Benjamin/Cummings, 1992.

[59] M. Batty, K. W. Axhausen, F. Giannotti, A. Pozdnoukhov, A. Bazzani, M. Wachowicz, G. Ouzounis, and Y. Portugali. Smart cities of the future. *The European Physical Journal Special Topics*, 214(1):481–518, Nov 2012.

[60] Edmon Begoli, Ian Goethert, and Kathryn Knight. A lakehouse architecture for the management and analysis of heterogeneous data for biomedical research and mega-biobanks. In *2021 IEEE International Conference on Big Data (Big Data)*, pages 4643–4651, 2021.

[61] Zohra Bellahsene, Angela Bonifati, and Erhard Rahm. *Schema Matching and Mapping*. Springer Publishing Company, Incorporated, 1st edition, 2011.

[62] Eline A Belt, Thomas Koch, and Elenna R Dugundji. Hourly forecasting of traffic flow rates using spatial temporal graph neural networks. *Procedia Computer Science*, 220:102–109, 2023.

[63] Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the dangers of stochastic parrots: Can language models be too big? In *Conference on Fairness, Accountability, and Transparency (FAccT)*, page 610–623. ACM, 2021.

[64] Philip A. Bernstein. Applying model management to classical meta data problems. In *Conference on Innovative Data Systems Research (CIDR)*, 2003.

[65] Philip A. Bernstein and Howard Ho. Model management and schema mappings: Theory and practice. In *Proc. VLDB Endow.*, pages 1439–1440. ACM, 2007.

[66] Luca Bertinetto, Romain Mueller, Konstantinos Tertikas, Sina Samangooei, and Nicholas A. Lord. Making Better Mistakes: Leveraging Class Hierarchies With Deep Networks . In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12503–12512, 2020.

[67] S. Bhattacharya and S. Mishra. Applications of machine learning for facies and fracture prediction using bayesian network theory and random forest: Case studies from the appalachian basin, usa. *Journal of Petroleum Science and Engineering*, 170:1005–1017, 2018.

[68] Devis Bianchini, Valeria De Antonellis, Massimiliano Garda, and Michele Melchiori. Smart city data modelling using semantic web technologies. In *IEEE International Smart Cities Conference (ISC2)*, pages 1–7, 2021.

[69] Simon Elias Bibri. The iot for smart sustainable cities of the future: An analytical framework for sensor-based big data applications for environmental sustainability. *Sustainable Cities and Society*, 38:230–253, 2018.

[70] Simon Elias Bibri and John Krogstie. The core enabling technologies of big data analytics and context-aware computing for smart sustainable cities: a review and synthesis. *Journal of Big Data*, 4(1):38, Nov 2017.

[71] Stefan Bischof, Athanasios Karapantelakis, Cosmin-Septimiu Nechifor, Amit P. Sheth, Alessandra Mileo, and Payam M. Barnaghi. Semantic modelling of smart city data. In *W3C*, 2014.

[72] Asim Biswal, Liana Patel, Siddarth Jha, Amog Kamsetty, Shu Liu, Joseph E. Gonzalez, Carlos Guestrin, and Matei Zaharia. Text2SQL is not Enough: Unifying AI and databases with TAG. *CORR*, 2024.

[73] Dominik Bork, Syed Juned Ali, and Ben Roelens. Conceptual modeling and artificial intelligence: A systematic mapping study. *arXiv preprint*, 2023.

[74] Peter Bosch, Sophie Jongeneel, Hans-Martin Neumann, Iglar Branislav, and Aapo Huovila. Recommendations for a smart city index. *Project deliverable, D3.3*, 2016.

[75] Peter Bosch, Sophie Jongeneel, Vera Rovers, Hans-Martin Neumann, Mimu Airaksinen, and Aapo Huovila. Citykeys indicators for smart city projects and smart cities. *Report*, 2017.

[76] Priyankar Bose, Sriram Srinivasan, William C. Sleeman, Jatinder Palta, Rishabh Kapoor, and Preetam Ghosh. A survey on recent named entity recognition and relationship extraction techniques on clinical texts. *Applied Sciences*, 11(18), 2021.

[77] Harry Brignull and Yvonne Rogers. Enticing people to interact with large public displays in public spaces. In *Human-Computer Interaction (INTERACT)*, 2003.

[78] Matthias Budde, Andrea Schankin, Julien Hoffmann, Marcel Danz, Till Riedel, and Michael Beigl. Participatory sensing or participatory nonsense? mitigating the effect of human error on data quality in citizen science. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 1(3), sep 2017.

[79] Nélio Cacho, Frederico Lopes, and Thaís Batista. Challenges to the development of smart city systems: A system-of-systems view. In *SBES 2017*, pages 244–249, 09 2017.

[80] Michael J Cafarella, Alon Halevy, Daisy Zhe Wang, Eugene Wu, and Yang Zhang. Webtables: exploring the power of tables on the web. *Proceedings of the VLDB Endowment*, 1(1):538–549, 2008.

[81] Hongming Cai, Boyi Xu, Lihong Jiang, and Athanasios V Vasilakos. Iot-based big data storage systems in cloud computing: perspectives and challenges. *IEEE Internet of Things Journal*, 4(1):75–87, 2016.

[82] Richard L Caldwell, Willett F Baldwin, James D Bargainer, James E Berry, George N Salaita, and Raymond W Sloan. Gamma-ray spectroscopy in well logging. *Geophysics*, 28(4):617–632, 1963.

[83] Riccardo Cappuzzo, Paolo Papotti, and Saravanan Thirumuruganathan. Creating embeddings of heterogeneous relational datasets for data integration tasks. In *International Conference on Management of Data (SIGMOD)*, pages 1335–1349. ACM, 2020.

[84] Paolo Cardullo and Rob Kitchin. Smart urbanism and smart citizenship: The neoliberal logic of 'citizen-focused' smart cities in europe. *Environment and Planning C: Politics and Space*, 37(5):813–830, 2019.

[85] Salvatore Carta, Alessandro Giuliani, Leonardo Piano, Alessandro Sebastian Podda, Livio Pompianu, and Sandro Gabriele Tiddia. Iterative Zero-Shot LLM Prompting for Knowledge Graph Construction. *arXiv preprint*, 2023.

[86] François Castagnos, Martin Mihelich, and Charles Dognin. A simple log-based loss function for ordinal text classification. In *Proceedings of the 29th International Conference on Computational Linguistics*, 2022.

[87] Rick Cattell. Scalable SQL and NoSQL data stores. *SIGMOD Record*, 39(4):12–27, 2010.

[88] C. Cattuto, M. Quaggiotto, A. Panisson, and A. Averbuch. Time-varying social networks in a graph database: a Neo4j use case. In *International Workshop on Graph Data Management Experiences and Systems (GRADES)*, pages 1–6. ACM, 2013.

[89] Everton Cavalcante, Nélio Cacho, Frederico Lopes, Thais Batista, and Flavio Oquendo. Thinking smart cities as systems-of-systems: A perspective study. In *International Workshop on Smart Cities*, SmartCities. ACM, 2016.

[90] Patricio Cerda, Gaël Varoquaux, and Balázs Kégl. Similarity encoding for learning with dirty categorical variables. *Machine Learning*, 107(8-10):1477–1494, 2018.

[91] Pablo Chamoso, Alfonso González-Briones, Sara Rodríguez, Juan M. Corchado, and Ramón Sanchez. Tendencies of technologies and platforms in smart cities: A state-of-the-art review. *Wireless Communications and Mobile Computing*, 2018:17, 2018.

[92] Fay Chang et al. Bigtable: A distributed storage system for structured data. *ACM Trans. Comput. Syst.*, 26(2), 2008.

[93] C Aggarwal Charu and K REDDY Chandan. *Data clustering: algorithms and applications*. Chapman and Hall/CRC Boca Raton, 2013.

[94] Artem Chebotko, Andrey Kashlev, and Shiyong Lu. A Big Data Modeling Methodology for Apache Cassandra. In *IEEE International Conference on Big Data (BigData)*, pages 238–245, 2015.

[95] Qi Chen, Wei Wang, Fangyu Wu, Suparna De, Ruili Wang, Bailing Zhang, and Xin Huang. A survey on an emerging area: Deep learning for smart city data. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 3(5):392–410, 2019.

[96] Yang Chen, Arturo Ardila-Gomez, and Gladys Frame. Achieving energy savings by intelligent transportation systems investments in the context of smart cities. *Transportation Research Part D: Transport and Environment*, 54:381–396, 2017.

[97] Zui Chen, Zihui Gu, Lei Cao, Ju Fan, Samuel Madden, and Nan Tang. Symphony: Towards natural language query answering over multi-modal data lakes. In *Conference on Innovative Data Systems Research (CIDR)*, 2023.

[98] Max Chevalier, Mohammed El Malki, Arlind Kopliku, Olivier Teste, and Ronan Tournier. Implementation of multidimensional databases in column-oriented NoSQL systems. In *19th East European Conference on Advances in Databases and Information Systems (ADBIS 2015)*, pages 79–91, 2015.

[99] Chicago. City of chicago. data portal.

[100] Kristina Chodorow. *MongoDB: The Definitive Guide*. O'Reilly Media, 2013.

[101] Paul F. Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[102] United 4 Smart Sustainable Cities. Collection methodology for key performance indicators for smart sustainable cities. *United 4*, 2017.

[103] European Comission. Communication from the commission to the european parliament, the council, the european economic and social committee and the committee of the regions. a european strategy for data, 2020.

[104] European Commission. Open data goldbook for data managers and data holders. practical guidebook for organizations wanting to publish open data. *EUData*, 2018.

[105] European Commission and Directorate-General for Environment. *Indicators for sustainable cities*. Publications Office, 2018.

[106] Sergio Consoli, Misael Mongiovic, Andrea G. Nuzzolese, Silvio Peroni, Valentina Presutti, Diego Reforgiato Recupero, and Daria Spampinato. A smart city data model based on semantics best practice and principles. In *Conference on World Wide Web (WWW)*, pages 1395–1400. ACM, 2015.

[107] Carlos Costa and Maribel Yasmina Santos. The suscity big data warehousing approach for smart cities. In *ACM*, IDEAS 2017, pages 264–273. ACM, 2017.

[108] World Council. World council on city data.

[109] M. Cruz-Ramírez, C. Hervás-Martínez, J. Sánchez-Monedero, and P.A. Gutiérrez. Metrics to guide a multi-objective evolutionary algorithm for ordinal classification. *Neurocomputing*, 135:21–31, 2014.

[110] Federico Cugurullo. *Frankenstein urbanism: eco, smart and autonomous cities, artificial intelligence and the end of the city*. Routledge, 2021.

[111] Thiago Pereira da Silva, Thais Batista, Frederico Lopes, Aluizio Rocha Neto, Flávia C. Delicato, Paulo F. Pires, and Atslands R. da Rocha. Fog computing platforms for smart city applications - a survey. *ACM Trans. Internet Technol.*, feb 2022.

[112] Mathieu d'Aquin, John Davies, and Enrico Motta. Smart cities' data: Challenges and opportunities for semantic technologies. *IEEE Internet Computing*, 19:66–70, 11 2015.

[113] City of New York Data, NYC Open. NYC Open Data.

[114] Ayona Datta. New urban utopias of postcolonial india: Entrepreneurial urbanization in dholera smart city, gujarat. *Dialogues in Human Geography*, 5(1):3–22, 2015.

[115] Islay Davies, Peter Green, Michael Rosemann, Marta Indulska, and Stan Gallo. How do practitioners use conceptual modeling in practice? *Data & Knowledge Engineering (DKE)*, 58(3):358–380, 2006.

[116] Ali Davoudian and Mengchi Liu. Big data systems: A software engineering perspective. *ACM Computing Surveys (CSUR)*, 53(5), 2020.

[117] Claudio de Lima and Ronaldo dos Santos Mello. A workload-driven logical design approach for NoSQL document databases. In *Int. Conference on Information Integration and Web-based Applications & Services (iiWAS)*, pages 73:1–73:10. ACM, 2015.

[118] Arthur de M. Del Esposte, Eduardo F.Z. Santana, Lucas Kanashiro, Fabio M. Costa, Kelly R. Braghetto, Nelson Lago, and Fabio Kon. Design and evaluation of a scalable smart city software platform with large-scale simulations. *Future Gener. Comput. Syst.*, 93(C):427–441, apr 2019.

[119] Christopher De Sa, Alex Ratner, Christopher Ré, Jaeho Shin, Feiran Wang, Sen Wu, and Ce Zhang. Deepdive: Declarative knowledge base construction. *International Conference on Management of Data (SIGMOD)*, 45(1):60–67, 2016.

[120] Jeffrey Dean and Sanjay Ghemawat. Mapreduce: Simplified data processing on large clusters. In *Symposium on Operating System Design and Implementation (OSDI)*, pages 137–150, 2004.

[121] Aoife Delaney and Rob Kitchin. Progress and prospects for data-driven coordinated management and emergency response: the case of ireland. *Territory, Politics, Governance*, 11(1):174–189, 2023.

[122] Yuri Demchenko, Paola Grosso, Cees De Laat, and Peter Membrey. Addressing big data issues in scientific data infrastructure. In *International Conference on Collaboration Technologies and Systems (CTS)*, pages 48–55. IEEE, 2013.

[123] Zikun Deng, Di Weng, Shuhan Liu, Yuan Tian, Mingliang Xu, and Yingcai Wu. A survey of urban visual analytics: Advances and future directions. *Computational Visual Media*, 9, 2023.

[124] Abu Dhabi, editor. *Automated Well Correlation using Machine Learning and Facial Recognition Techniques*, International Petroleum Exhibition and Conference, 11 2020.

[125] AnHai Doan, Alon Y. Halevy, and Zachary G. Ives. *Principles of Data Integration*. Morgan Kaufmann, 2012.

[126] Xin Luna Dong and Theodoros Rekatsinas. Data integration and machine learning: A natural synergy. *Proc. VLDB Endow.*, 11(12):2094–2097, August 2018.

[127] Nicola Dragoni, Saverio Giallorenzo, Alberto Lluch Lafuente, Manuel Mazzara, Fabrizio Montesi, Ruslan Mustafin, and Larisa Safina. *Microservices: Yesterday, Today, and Tomorrow*, pages 195–216. Springer International Publishing, Cham, 2017.

[128] Wenjie Du, David Côté, and Yan Liu. Saits: Self-attention-based imputation for time series. *Expert Systems with Applications*, 219:119619, June 2023.

[129] Carmen Echebarria, Jose M. Barrutia, and Itziar Aguado-Moralejo. The smart city journey: a systematic review and future research agenda. *Innovation: The European Journal of Social Science Research*, 34(2):159–201, 2021.

[130] David Eckhoff and Isabel Wagner. Privacy in the smart city-applications, technologies, challenges, and solutions. *IEEE Communications Surveys & Tutorials*, 20(1):489–516, 2018.

[131] J. Eidsvik, T. Mukerji, and P. Switzer. Estimation of geological attributes from a well log: An application of hidden markov chains. *Mathematical Geology*, 36:379–397, 2004.

[132] Figure Eight. Figure eight. the essential high-quality data annotation platform.

[133] Karim El Hage, Adel Remadi, Yasmina Hobeika, Ruining Ma, Victor Hong, and Francesca Bugiotti. A multi-source graph database to showcase a

recommender system for dyslexic students. In *IEEE International Conference on Big Data (BigData)*, pages 3134–3138, 2023.

[134] Ahmed Eldawy, Vagelis Hristidis, Saheli Ghosh, Majid Saeedan, Akil Sevim, A.B. Siddique, Samriddhi Singla, Ganesh Sivaram, Tin Vu, and Yaming Zhang. Beast: Scalable exploratory analytics on spatio-temporal data. In *ACM International Conference on Information & Knowledge Management (CIKM)*, page 3796–3807. ACM, 2021.

[135] Bibri Simon Elias, Allam Zaheer, and Krogstie John. The metaverse as a virtual form of data-driven smart urbanism: platformization and its underlying processes, institutional dimensions, and disruptive impacts. *Computational Urban Science*, 2, 2022.

[136] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD*, pages 226–231, 1996.

[137] Adriana Eugene, Naomi Alpert, Wil Lieberman-Cribbin, and Emanuela Taioli. Using nyc 311 call center data to assess short-and long-term needs following hurricane sandy. *Disaster Medicine and Public Health Preparedness*, 16(4):1447–1451, 2022.

[138] Eric Evans. *Domain-Driven Design*. Addison-Wesley, 2003.

[139] Chinedu Pascal Ezenkwu, John Guntoro, Andrew Starkey, Vahid Vaziri, and Maurillio Addario. Automated well-log pattern alignment and depth-matching techniques: An empirical review and recommendations. *Petrophysics - The SPWLA Journal*, 64(01):115–129, 02 2023.

[140] Raul Castro Fernandez, Aaron J. Elmore, Michael J. Franklin, Sanjay Krishnan, and Chenhao Tan. How large language models will disrupt data management. *Proc. VLDB Endow.*, 16(11):3302–3309, jul 2023.

[141] Raul Castro Fernandez, Peter R. Pietzuch, Jay Kreps, Neha Narkhede, Jun Rao, Joel Koshy, Dong Lin, Chris Riccomini, and Guozhang Wang. Liquid: Unifying nearline and offline big data integration. In *Conference on Innovative Data Systems Research*, 2015.

[142] FIWARE. Fiware smart cities.

[143] Daniela Florescu and Donald Kossmann. Storing and querying XML data using an RDMBS. *IEEE Data Eng. Bull.*, 22(3):27–34, 1999.

[144] European Innovation Partnership for Smart Cities & Communities (EIP-SCC). Eip-scc urban platform management framework, enabling cities to maximize value from city data. *EIP-SCC*, 2016.

[145] The United for Smart Sustainable Cities. Redefining smart city platforms:setting the stage for minimal interoperability mechanisms. a u4ssc deliverable on city platforms, 2022.

[146] Apurva Gandhi, Yuki Asada, Victor Fu, Advitya Gemawat, Lihao Zhang, Rathijit Sen, Carlo Curino, Jesús Camacho-Rodríguez, and Matteo Interlandi. The tensor data platform: Towards an ai-centric database system. In *Conference on Innovative Data Systems Research (CIDR)*, 2023.

[147] Antonio Garmendia, Dominik Bork, Martin Eisenberg, Thiago do Nascimento Ferreira, Marouane Kessentini, and Manuel Wimmer. Leveraging artificial intelligence for model-based software analysis and design. In *Optimising the Software Development Process with Artificial Intelligence*, pages 93–117. Springer, 2023.

[148] Gartner. Market guide for smart city operations management platforms and ecosystems, 2015.

[149] Lisa Gaudette and Nathalie Japkowicz. Evaluation methods for ordinal classification. In Yong Gao and Nathalie Japkowicz, editors, *Advances in Artificial Intelligence*, pages 207–210, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[150] Aditya Gaur, Bryan Scotney, Gerard Parr, and Sally McClean. Smart city architecture and its applications based on iot. *Procedia Computer Science*, 52:1089 – 1094, 2015. International Conference on Sustainable Energy Information Technology (SEIT).

[151] Geomesa. Geomesa.

[152] Ammar Gharaibeh, Mohammad A. Salahuddin, Sayed Jahed Hussini, Abdallah Khreishah, Issa Khalil, Mohsen Guizani, and Ala Al-Fuqaha. Smart cities: A survey on data management, security, and enabling technologies. *IEEE Communications Surveys & Tutorials*, 19(4):2456–2501, 2017.

[153] Rudolf Giffinger, Christian Fertner, Hans Kramar andRobert Kalasek, Nataša Pichler-Milanović, and Evert Meijers. Smart cities: Ranking of european medium-sized cities. *Smart Cities*, 2007.

[154] Behzad Golshan, Alon Halevy, George Mihaila, and Wang-Chiew Tan. Data integration: After the teenage years. In *International Conference on Management of Data (SIGMOD)*, page 101–106. ACM, 2017.

[155] Google. Crowdsource by Google.

[156] Salvatore Greco, Alessio Ishizaka, Menelaos Tasiou, and Gianpiero Torrisi. On the methodological framework of composite indices: A review

of the issues of weighting, aggregation, and robustness. *Social Indicators Research*, 141(1):61–94, Jan 2019.

[157] G. Gröger, T.H. Kolbe, C. Nagel, and K.H. Häfele. Ogc city geography markup language (citygml) encoding standard. OGC Standard OGC 12-019 Open Geospatial Consortium, 2012, 2012. 35.01.01; LK 01.

[158] Thomas R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199 – 220, 1993.

[159] Michael Gubanov, Manju Priya, and Maksim Podkorytov. Intellilight: A flashlight for large-scale dark structured data, 2017.

[160] Abhimanyu Gupta, Geert Poels, and Palash Bera. Generating multiple conceptual models from behavior-driven development scenarios. *Data & Knowledge Engineering (DKE)*, 145:102141, 2023.

[161] Rahul Gupta, Alon Halevy, Xuezhi Wang, Steven Euijong Whang, and Fei Wu. Biperpedia: An ontology for search applications. *Proceedings of the VLDB Endowment*, 7(7):505–516, 2014.

[162] Ralf Hartmut Güting and Markus Schneider. *Moving Object Databases*. Morgan Kaufmann Publishers, 2005.

[163] Hadi Habibzadeh, Cem Kaptan, Tolga Soyata, Burak Kantarci, and Azzedine Boukerche. Smart city system design: A comprehensive study of the application and data planes. *ACM Comput. Surv.*, 52(2), May 2019.

[164] Hadi Habibzadeh, Tolga Soyata, Burak Kantarci, Azzedine Boukerche, and Cem Kaptan. Sensing, communication and security planes: A new challenge for a smart city system design. *Computer Networks*, 144:163–200, 2018.

[165] Hadoop. Spatialhadoop.

[166] Hadoop. St-hadoop.

[167] Jean-Luc Hainaut. The transformational approach to database engineering. In *GTTSE, LNCS 4143*, pages 95–143. Springer, 2006.

[168] Alon Halevy, Yejin Choi, Avrilia Floratou, Michael J. Franklin, Natasha Noy, and Haixun Wang. Will llms reshape, supercharge, or kill data science? *Proc. VLDB Endow.*, 16(12):4114–4115, aug 2023.

[169] Alon Halevy and Jane Dwivedi-Yu. Learnings from data integration for augmented language models. *arXiv preprint*, 2023.

[170] Alon Y. Halevy, Anand Rajaraman, and Joann J. Ordille. Data integration: The teenage years. In *Proc. VLDB Endow.*, pages 9–16. ACM, 2006.

[171] Dame Wendy Hall and Jérôme Pesenti. Growing the artificial intelligence industry in the uk. *GOV*, 2017.

[172] R E Hall, B Bowerman, J Braverman, J Taylor, H Todosow, and U Von Wimmersperg. The vision of a smart city. *GOV*, 9 2000.

[173] Sophia Hamer, Jennifer Sleeman, and Ivanka Stajner. Forecast-aware model driven lstm. *arXiv preprint arXiv:2303.12963*, 2023.

[174] Michael Hamrah. Data Modeling at Scale: MongoDB + Mongoid, Callbacks, and Denormalizing Data for Efficiency. http://blog.michaelhamrah.com/2011/08/data-modeling-at-scale-mongodb-mongoid-callbacks-and-denormalizing-data-for-ef 2011. (Accessed February, 2016).

[175] Jack Hardinges. What is a data trust?, 2018.

[176] Jack Hardinges and Peter Wells. Defining a data trust, 2018.

[177] C. Harrison, B. Eckman, R. Hamilton, P. Hartswick, J. Kalagnanam, J. Paraszczak, and P. Williams. Foundations for smarter cities. *IBM Journal of Research and Development*, 54(4):1–16, 2010.

[178] Guy Harrison. *Next Generation Databases: NoSQL, NewSQL, and Big Data*. Apress, 2016.

[179] Ibrahim Abaker Targio Hashem, Victor Chang, Nor Badrul Anuar, Kayode Adewole, Ibrar Yaqoob, Abdullah Gani, Ejaz Ahmed, and Haruna Chiroma. The role of big data in smart city. *International Journal of Information Management*, 36(5):748 – 758, 2016.

[180] Tali Hatuka, Toch Eran, Birnhack Michael, and Hadas Zur. *The Digital City: Critical Dimensions in Implementing the Smart City*. SSRN, 2020.

[181] J. He, A. D. La Croix, J. Wang, W. Ding, and J. R. Underschultz. Using neural networks and the markov chain approach for facies analysis and prediction from well logs in the precipice sandstone and evergreen formation, surat basin, australia. *Marine and Petroleum Geology*, 101:410–427, 2019.

[182] Wei He, Wanqiang Li, and Peidong Deng. Legal governance in the smart cities of china: Functions, problems, and solutions. *Sustainability*, 14(15):9738, 2022.

[183] Pat Helland. Life beyond distributed transactions: an apostate's opinion. In *Conference on Innovative Data Systems Research (CIDR)*, pages 132–141, 2007.

[184] Amr Hendy, Mohamed Abdelrehim, Amr Sharaf, Vikas Raunak, Mohamed Gabr, Hitokazu Matsushita, Young Jin Kim, Mohamed Afify, and Hany Awadalla. How Good Are GPT Models at Machine Translation? A Comprehensive Evaluation. *arXiv preprint*, 2023.

[185] Roberto Henry Herrera and Mirko van der Baan. Automated seismic-to-well ties?, 2012.

[186] RC Hertzog and RE Plasek. Neutron-excited gamma-ray spectrometry for well logging. *IEEE Transactions on Nuclear Science*, 26(1):1558–1567, 1979.

[187] Moditha Hewasinghage, Nacéra Bennacer Seghouani, and Francesca Bugiotti. Modeling strategies for storing data in distributed heterogeneous NoSQL databases. In *International Conference on Conceptual Modeling (ER)*, volume 11157, pages 488–496. Springer, 2018.

[188] Arne Hintz, Lina Dencik, and Karin Wahl-Jorgensen. Digital citizenship and surveillance | digital citizenship and surveillance society — introduction. *International Journal of Communication*, 11(0), 2017.

[189] Taisei Hirakawa, Keisuke Maeda, Takahiro Ogawa, Satoshi Asamizu, and Miki Haseyama. Analysis of social trends related to covid-19 pandemic utilizing social media data. In *Global Conference on Consumer Electronics (GCCE)*, pages 43–44. IEEE, 2021.

[190] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[191] Robert G. Hollands. Critical interventions into the corporate smart city. *Cambridge Journal of Regions, Economy and Society*, 8(1):61–77, 08 2014.

[192] Ali Reza Honarvar and Ashkan Sami. Towards sustainable smart city by particulate matter prediction using urban big data, excluding expensive air pollution infrastructures. *Big data research*, 17:56–65, 2019.

[193] Simo Hosio, Vassilis Kostakos, Hannu Kukka, Marko Jurmu, Jukka Riekki, and Timo Ojala. From school food to skate parks in a few clicks: Using public displays to bootstrap civic engagement of the young. In Judy Kay, Paul Lukowicz, Hideyuki Tokuda, Patrick Olivier, and Antonio Krüger, editors, *Pervasive Computing*, pages 425–442. Springer, 2012.

[194] Le Hou, Chen-Ping Yu, and Dimitris Samaras. Squared earth mover's distance-based loss for training deep neural networks. *CoRR*, 2016.

[195] Jeremy Howard et al. fastai. `https://github.com/fastai/fastai`, 2018.

[196] HPCC. Hpcc systems.

[197] HPCC. Taming the Data Lake: The HPCC Systems Open Source Big Data Platform.

[198] Aapo Huovila, Peter Bosch, and Miimu Airaksinen. Comparative analysis of standardized indicators for smart sustainable cities: What indicators and standards to use and when? *Cities*, 89:141 – 153, 2019.

[199] Sifat Ibtisum, S M Atikur Rahman, and s. M. Saokat Hossain. Comparative analysis of mapreduce and apache tez performance in multinode clusters with data compression. *World Journal of Advanced Research and Reviews*, 20:519–526, 12 2023.

[200] Sergio Ilarri, Eduardo Mena, and Arantza Illarramendi. Location-dependent query processing: Where we are and where we are heading. *ACM Comput. Surv.*, 42(3), mar 2010.

[201] European Telecommunication Standards Institute. Etsi ts103463 key performance indicators for sustainable digital multiservice cities. technical specification v1.1.1 (2017-07), 2017.

[202] European Telecommunication Standards Institute. Context information management (cim); information model (modo), etsi gs cim 006 v1.1.1 (2019-07), group specification, 2019.

[203] European Telecommunication Standards Institute. Context information management (cim);ngsi-ld; guidelines for the deployment of smart city and communities data platforms. etsi gr cim 020 v1.1.1 (2022-12), group report, 2022.

[204] European Telecommunication Standards Institute. Cross-cutting context information management (cim); ngsi-ld api, etsi gs cim 009 v1.6.1 (2022-08), group specification, 2022.

[205] Open Data Institute. Mapping the wide world of data sharing.

[206] Hassan Ismail Fawaz, Benjamin Lucas, Germain Forestier, Charlotte Pelletier, Daniel F. Schmidt, Jonathan Weber, Geoffrey I. Webb, Lhassane Idoumghar, Pierre-Alain Muller, and François Petitjean. Inceptiontime: Finding alexnet for time series classification. *Data Mining and Knowledge Discovery*, 34(6):1936–1962, September 2020.

[207] ISO. International organization for standardization.

[208] ISO219722020. Iso/iec 21972:2020 information technology - upper level ontology for smart city indicators.

[209] ISO37120. International standard iso 37120, sustainable cities and communities — indicators for city services and quality of life, 2018.

[210] ISO37122. International standard iso 37122, sustainable cities and communities - indicators for smart cities, 2019.

[211] ISO50871. Iso/iec prf 5087-1 information technology - city data model - part 1: Foundation level concepts.

[212] Shrainik Jain, Dominik Moritz, Daniel Halperin, Bill Howe, and Ed Lazowska. SQLShare: Results from a multi-year SQL-as-a-Service experiment. In *International Conference on Management of Data (SIGMOD)*, pages 281–293, 2016.

[213] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023.

[214] Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mixtral of experts. *arXiv preprint*, 2024.

[215] Huaxiong Jiang, Stan Geertman, and Patrick Witte. The contextualization of smart city technologies: An international comparison. *Journal of Urban Management*, 12(1):33–43, 2023.

[216] Sihang Jiang, Jiaqing Liang, Yanghua Xiao, Haihong Tang, Haikuan Huang, and Jun Tan. Towards the completion of a domain-specific knowledge base with emerging query terms. In *IEEE International Conference on Data Engineering*, pages 1430–1441. IEEE, 2019.

[217] Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y. Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, and Qingsong Wen. Time-llm: Time series forecasting by reprogramming large language models, 2024.

[218] Alekh Jindal, Shi Qiao, Sathwik Reddy Madhula, Kanupriya Raheja, and Sandhya Jain. Turning databases into generative ai machines. In *Conference on Innovative Data Systems Research (CIDR)*, 2024.

[219] Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351, 10 2017.

[220] Collin Joyce, Benjamin Nemoz, Raiza Bastidas, Bryan Briney, and Dennis R. Burton. Longitudinal analysis of drift in the circulating human antibody repertoire over four years. *bioRxiv*, 2025.

[221] Kyung Hwa Jung, Zachary Pitkowsky, Kira Argenio, James W Quinn, Jean-Marie Bruzzese, Rachel L Miller, Steven N Chillrud, Matthew Perzanowski, Jeanette A Stingone, and Stephanie Lovinsky-Desir. The effects of the historical practice of residential redlining in the united states on recent temporal trends of air pollution near new york city schools. *Environment international*, 169:107551, 2022.

[222] A. Kalinowski, D. Datta, and Y. An. A scalable approach to aligning natural language and knowledge graph representations: Batched information guided optimal transport. In *IEEE International Conference on Big Data (BigData)*, pages 383–392, 2023.

[223] Fazle Karim, Somshubra Majumdar, Houshang Darabi, and Shun Chen. Lstm fully convolutional networks for time series classification. *IEEE Access*, 6:1662–1669, 2018.

[224] Ilya Katsov. NoSQL data modeling techniques. Highly Scalable Blog, `https://highlyscalable.wordpress.com/2012/03/01/nosql-data-modeling-techniques/`, 2012. accessed February 2016.

[225] T. Kaufmann, P. Weng, V. Bengs, and E. Hüllermeier. A survey of reinforcement learning from human feedback. *arXiv preprint arXiv:2312.14925*, 2023.

[226] Moe Kayali, Anton Lykov, Ilias Fountalis, Nikolaos Vasiloglou, Dan Olteanu, and Dan Suciu. CHORUS: Foundation Models for Unified Data Discovery and Exploration. *arXiv preprint*, 2023.

[227] Moe Kayali, Fabian Wenz, Nesime Tatbul, and Çağatay Demiralp. Mind the data gap: Bridging llms to enterprise data integration, 2024.

[228] Latif U. Khan, Ibrar Yaqoob, Nguyen H. Tran, S. M. Ahsan Kazmi, Tri Nguyen Dang, and Choong Seon Hong. Edge-computing-enabled

smart cities: A comprehensive survey. *IEEE Internet of Things Journal*, 7(10):10200–10232, 2020.

[229] Amandeep Khurana. Introduction to HBase Schema Design. *;login: The Usenix magazine*, 37(5):29–36, 2012.

[230] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.

[231] Rob Kitchin. The real-time city? big data and smart urbanism. *GeoJournal*, 79(1):1–14, Feb 2014.

[232] Rob Kitchin, Tracey P. Lauriault, and Gavin McArdle. Knowing and governing cities through urban indicators, city benchmarking and real-time dashboards. *Regional Studies, Regional Science*, 2(1):6–28, 2015.

[233] Rob Kitchin and Niamh Moore-Cherry. Fragmented governance, the urban data ecosystem and smart city-regions: the case of metropolitan boston. *Regional Studies*, 55(12):1913–1923, 2021.

[234] Martin Kleppmann. *Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems*. O Reilly Media, 2017.

[235] Bram Klievink, Haiko van der Voort, and Wijnand Veeneman. Creating value through data collaboratives. *Information Polity*, 23(4):379–397, 2018.

[236] Jason Koh, Sandeep Sandha, Bharathan Balaji, Daniel Crawl, Ilkay Altintas, Rajesh E. Gupta, and Mani B. Srivastava. Data hub architecture for smart cities. In *Conference on Embedded Network Sensor Systems*, pages 77:1–77:2, 2017.

[237] Anusha Kola, Harshal More, Sean Soderman, and Michael Gubanov. Generating unified famous objects (ufos) from the classified object tables. In *IEEE International Conference on Big Data (BigData)*, pages 4771–4773. IEEE, 2017.

[238] Andreas Komninos, Jeries Besharat, Denzil Ferreira, John Garofalakis, and Vassilis Kostakos. Where's everybody? comparing the use of heatmaps to uncover cities' tacit social context in smartphones and pervasive displays. *Information Technology & Tourism*, 17:399–427, 2017.

[239] Vassilis Kostakos, Jakob Rogstadius, Denzil Ferreira, Simo Hosio, and Jorge Goncalves. *Human Sensors*, page 69–92. Understanding Complex Systems. Springer International Publishing, 2017.

[240] Jay Kreps. Questioning the lambda architecture, 2014.

[241] M. Kulmala, T. V. Kokkonen, J. Pekkanen, S. Paatero, T. Petäjä, V.-M. Kerminen, and A. Ding. Opinion: Gigacity – a source of problems or the new way to sustainable development. *Atmospheric Chemistry and Physics*, 21(10):8313–8322, 2021.

[242] Sidewalk Labs. The digital innovation plan. *MIDP*, 2019.

[243] Chun Sing Lai, Youwei Jia, Zhekang Dong, Dongxiao Wang, Yingshan Tao, Qi Hong Lai, Richard T. K. Wong, Ahmed F. Zobaa, Ruiheng Wu, and Loi Lei Lai. A review of technical standards for smart cities. *Clean Technologies*, 2(3):290–310, 2020.

[244] Jin Lai, Yang Su, Lu Xiao, Fei Zhao, Tianyu Bai, Yuhang Li, Hongbin Li, Yuyue Huang, Guiwen Wang, and Ziqiang Qin. Application of geophysical well logs in solving geologic issues: Past, present and future prospect. *Geoscience Frontiers*, 15(3):101779, 2024.

[245] J LeFevre, J. Sankaranarayanan, H. Hacigümüs, J. Tatemura, N Polyzotis, and M.J. Carey. MISO: souping up big data query processing with a multistore system. In *International Conference on Management of Data (SIGMOD)*, 2014.

[246] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, and D. Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474, 2020.

[247] Huahang Li, Longyu Feng, Shuangyin Li, Fei Hao, Chen Jason Zhang, Yuanfeng Song, and Lei Chen. On leveraging large language models for enhancing entity resolution. *arXiv preprint*, 2024.

[248] Peng Li, Peng Sun, Qiong Tang, Hang Yan, Yuanbin Wu, Xuanjing Huang, and Xipeng Qiu. Codeie: Large code generation models are better few-shot information extractors. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 15339–15353, 2023.

[249] Yuliang Li, Jinfeng Li, Yoshihiko Suhara, AnHai Doan, and Wang-Chiew Tan. Deep entity matching with pre-trained language models. *Proc. VLDB Endow.*, 14(1):50–60, sep 2020.

[250] Z. Li, Z. Yang, and M. Wang. Reinforcement learning with human feedback: Learning dynamic choices via pessimism. *arXiv preprint arXiv:2305.18438*, 2023.

[251] T Warren Liao. Clustering of time series data—a survey. *Pattern recognition*, 38(11):1857–1874, 2005.

[252] Chiehyeon Lim, Kwang-Jae Kim, and Paul P. Maglio. Smart cities with big data: Reference models, challenges, and considerations. *Cities*, 82:86 – 99, 2018.

[253] Wan Shen Lim, Matthew Butrovich, William Zhang, Andrew Crotty, Lin Ma, Peijing Xu, Johannes Gehrke, and Andrew Pavlo. Database gyms. In *Conference on Innovative Data Systems Research (CIDR)*, 2023.

[254] Peng Lin, Ji-gen Xia, Qiu-yuan Hou, Yong-li Ji, and Chen Li. An intelligent depth correction method for logging curves based on pearson correlation coefficient and dtw. In Jia'en Lin, editor, *Proceedings of the International Field Exploration and Development Conference*, pages 102–114. Springer Nature Singapore, 2024.

[255] Dianbo Liu, Ricky Sahu, Vlad Ignatov, Dan Gottlieb, and Kenneth Mandl. High performance computing on flat fhir files created with the new smart/hl7 bulk data access standard. *AMIA Symposium*, 2019:592–596, 03 2020.

[256] Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. What makes good in-context examples for GPT-3? In *Proceedings of Deep Learning Inside Out (DeeLIO)*. ACL, May 2022.

[257] London. Find open data. data portal.

[258] London. London Datastore – Greater London Authority.

[259] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440, 2015.

[260] Roman Lukyanenko, Arturo Castellanos, Jeffrey Parsons, Monica Chiarini Tremblay, and Veda C. Storey. Using conceptual modeling to support machine learning. In *International Conference on Advanced Information Systems Engineering (CAiSE)*, volume 350, pages 170–181. Springer, 2019.

[261] Anna Luusua, Johanna Ylipulli, and Emilia Rönkkö. Nonanthropocentric design and smart cities in the anthropocene. *it - Information Technology*, 59(6):295–304, 2017.

[262] Meiyi Ma, Sarah M. Preum, Mohsin Y. Ahmed, William Tärneberg, Abdeltawab Hendawi, and John A. Stankovic. Data sets, modeling, and decision making in smart cities: A survey. *ACM Trans. Cyber-Phys. Syst.*, 4(2), nov 2019.

[263] Wolfgang Maass and Veda C. Storey. Pairing conceptual modeling with machine learning. *Data & Knowledge Engineering (DKE)*, 134:101909, 2021.

[264] Martino Maggio, Francesco Arigliano, Ömer Özdemir, José Manuel Cantera, Eunah Kim, Ignacio Elicegui Maestro, Andrea Gaglione, and Angelo Capossele. Reference architecture for iot enabled smart cities, update. *Europa*, 2018.

[265] P Makkaroon, DQ Tong, Y Li, EJ Hyer, P Xian, S Kondragunta, PC Campbell, Y Tang, BD Baker, MD Cohen, et al. Development and evaluation of a north america ensemble wildfire air quality forecast: Initial application to the 2020 western united states "gigafire". *Journal of Geophysical Research: Atmospheres*, 128(22):e2022JD037298, 2023.

[266] Claudia Malzer and Marcus Baum. A hybrid approach to hierarchical density-based cluster selection. In *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*. IEEE, September 2020.

[267] Hug March and Ramon Ribera-Fumaz. Smart contradictions: The politics of making barcelona a self-sufficient city. *European Urban and Regional Studies*, 23(4):816–830, 2016.

[268] J.L. Mari, P. Gaudiani, and J. Delay. Characterization of geological formations by physical parameters obtained through full waveform acoustic logging. *Physics and Chemistry of the Earth, Parts A/B/C*, 36(17):1438–1449, 2011. Clays in Natural & Engineered Barriers for Radioactive Waste Confinement.

[269] Nathan Marz and James Warren. *Big Data: Principles and Best Practices of Scalable Realtime Data Systems*. Manning Publications Co., USA, 1st edition, 2015.

[270] Audrey L. Mayer. Strengths and weaknesses of common sustainability indices for multidimensional systems. *Environment International*, 34(2):277 – 291, 2008.

[271] Peter McBrien and Alexandra Poulovassilis. A uniform approach to inter-model transformations. In *CAiSE Conference, LNCS 1626*, pages 333–348, 1999.

[272] Colin McFarlane and Ola Söderström. On alternative smart cities. *City*, 21(3-4):312–328, 2017.

[273] Hassan Mehmood, Ekaterina Gilman, Marta Cortes, Panos Kostakos, Andrew Byrne, Katerina Valta, Stavros Tekes, and Jukka Riekki. Implementing big data lake for heterogeneous data sources. In *International*

Conference on Data Engineering Workshops (ICDEW), pages 37–44. IEEE, 2019.

[274] A Middleton and PDLR Solutions. Hpcc systems: Introduction to hpcc (high-performance computing cluster). *White paper, LexisNexis Risk Solutions*, 2011.

[275] Nandana Mihindukulasooriya, Sanju Tiwari, Carlos F. Enguix, and Kusum Lata. Text2kgbench: A benchmark for ontology-driven knowledge graph generation from text. In *The Semantic Web*, pages 247–265, 2023.

[276] Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 11048–11064. ACL, dec 2022.

[277] Michael J. Mior, Kenneth Salem, Ashraf Aboulnaga, and Rui Liu. Nose: Schema design for nosql applications. In *IEEE International Conference on Data Engineering*, pages 181–192, 2016.

[278] C. Mohan. History repeats itself: sensible and NonsenSQL aspects of the NoSQL hoopla. In *EDBT*, pages 11–16, 2013.

[279] MongoDB Inc. MongoDB. http://www.mongodb.org. accessed February 2016.

[280] Luca Mora, Roberto Bolici, and Mark Deakin. The first two decades of smart-city research: A bibliometric analysis. *Journal of Urban Technology*, 24(1):3–27, 2017.

[281] Vaia Moustaka, Athena Vakali, and Leonidas G. Anthopoulos. A systematic review for smart city data analytics. *ACM Computing Surveys*, 51(5), 2018.

[282] Georgios Mylonas, Athanasios Kalogeras, Georgios Kalogeras, Christos Anagnostopoulos, Christos Alexakos, and Luis Muñoz. Digital twins from smart manufacturing to smart cities: A survey. *IEEE Access*, 9:143222–143249, 2021.

[283] Meinard Müller. *Dynamic Time Warping*, pages 69–84. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

[284] Taewoo Nam and Theresa A. Pardo. Conceptualizing smart city with dimensions of technology, people, and institutions. In *Proceedings of the 12th Annual International Digital Government Research Conference: Digital*

Government Innovation in Challenging Times, dg.o '11, pages 282–291. ACM, 2011.

[285] Avanika Narayan, Ines Chami, Laurel Orr, and Christopher Ré. Can Foundation Models Wrangle Your Data? *Proc. VLDB Endow.*, 16(4):738–746, dec 2022.

[286] Michela Nardo, Michaela Saisana, Andrea Saltelli, Stefano Tarantola, Anders Hoffman, and Enrico Giovannini. *Handbook on Constructing Composite Indicators and User Guide*, volume 2005. OECD, 09 2008.

[287] Fedelucio Narducci, Marco Comerio, Carlo Batini, and Marco Castelli. A similarity-based framework for service repository integration. *Data & Knowledge Engineering (DKE)*, 106:18–35, 2016.

[288] United Nations. World urbanisation prospects. the 2014 revision, 2015.

[289] A. Nayak, Anil Poriya, and Dikshay Poojary. Type of nosql databases and its comparison with relational databases. *International Journal of Applied Information Systems*, 5(4):16–19, 2013.

[290] Paolo Neirotti, Alberto De Marco, Anna Corinna Cagliano, Giulio Mangano, and Francesco Scorrano. Current trends in smart city initiatives: Some stylised facts. *Cities*, 38:25–36, 2014.

[291] Neo4j. K-means clustering.

[292] Neo4j. Neo4j.

[293] Neo4j. Neo4j cypher query language.

[294] NEXLA. An introduction to big data formats understanding avro, parquet, and orc. In *NEXLA White paper*, pages 1–12, 2018.

[295] Jan Kristof Nidzwetzki and Ralf Hartmut Güting. Distributed secondo: A highly available and scalable system for spatial data processing. In Christophe Claramunt, Markus Schneider, Raymond Chi-Wing Wong, Li Xiong, Woong-Kee Loh, Cyrus Shahabi, and Ki-Joune Li, editors, *Advances in Spatial and Temporal Databases*, pages 491–496, Cham, 2015. Springer International Publishing.

[296] Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers, 2023.

[297] N. Nishikawa, S. Fujiwara, Y. Hayamizu, and K. Goda. Physical database design for manufacturing business analytics. In *IEEE International Conference on Big Data (BigData)*, pages 1793–1802, dec 2023.

[298] J. R. Norris. *Markov Chains*, volume 2 of *Cambridge Series in Statistical and Probabilistic Mathematics*. Cambridge University Press, 1998.

[299] Municipality of Copenhagen and Capital Region of Denmark. City data exchange - lessons learned from a public/private data collaboration. *Municipality*, 2018.

[300] Department of Natural Resources. Colorado energy and carbon management comission. https://ecmc.state.co.us/data.html#/cogis.

[301] Ignacio Oguiza. tsai - a state-of-the-art deep learning library for time series and sequential data. Github, 2023.

[302] Kieron O'Hara. Data trusts: Ethics, architecture and governance for trustworthy data stewardship. *Web Science Institute White Papers*, 2019.

[303] Tomoya Ohyama, Kazunori Hanyu, Masayuki Tani, and Momoka Nakae. Investigating crime harm index in the low and downward crime contexts: a spatio-temporal analysis of the japanese crime harm index. *Cities*, 130:103922, 2022.

[304] Tal Olier. Database design using key-value tables. http://www.devshed.com/c/a/mysql/database-design-using-key-value-tables/, 2006. accessed February 2016.

[305] Aiko Oliveira, Eduardo Nascimento, João Pinheiro, Caio Viktor S. Avila, Gustavo Coelho, Lucas Feijó, Yenier Izquierdo, Grettel García, Luiz André P. Paes Leme, Melissa Lemos, and Marco A. Casanova. Small, medium, and large language models for text-to-sql. In Wolfgang Maass, Hyoil Han, Hasan Yasar, and Nick Multari, editors, *Conceptual Modeling*, pages 276–294. Springer Nature Switzerland, 2025.

[306] Harley Vera Olivera, Maristela Holanda, Valeria Guimarâes, Fernanda Hondo, and Wagner Boaventura. Data modeling for NoSQL document-oriented databases. In *Annual Int. Symposium on Information Management and Big Data (SIMBig)*, volume 1478 of *CEUR Workshop Proceedings*, pages 129–135, 2015.

[307] Frederik Olsen, Calogero Schillaci, Mohamed Ibrahim, and Aldo Lipani. Borough-level covid-19 forecasting in london using deep learning techniques and a novel mse-moran's i loss function. *Results in Physics*, 35:105374, 2022.

[308] Oracle. Oracle NoSQL Database. http://www.oracle.com/us/products/database/nosql/. accessed February 2016.

[309] Francis Ostermeijer, Hans Koster, Leonardo Nunes, and Jos van Ommeren. Citywide parking policy and traffic: Evidence from amsterdam. *Journal of Urban Economics*, 128:103418, 2022.

[310] Ahmed Oussous, Fatima-Zahra Benjelloun, Ayoub Ait Lahcen, and Samir Belfkih. Big data technologies: A survey. *Journal of King Saud University-Computer and Information Sciences*, 30(4):431–448, 2018.

[311] S. Ouzineb and S. Wlodarczyk. Leveraging analog wells to fine-tune ai foundation models for well log prediction. In *SPE Western Regional Meeting*, page D031S006R006. SPE, April 2025.

[312] Laura Pareja Prieto. Introducing object storage in hadoop ecosystem. Technical report, Hadoop, 2022.

[313] Diego Pasqualin, Giovanni Souza, Eduardo Luis Buratti, Eduardo Cunha de Almeida, Marcos Didonet Del Fabro, and Daniel Weingaertner. A case study of the aggregation query model in read-mostly NoSQL document stores. In *20th Int. Database Engineering & Applications Symposium (IDEAS '16)*, IDEAS 2016, pages 224–229. ACM, 2016.

[314] Eric Paulos, Ian Smith, and R Honicky. Participatory urbanism. *urbanatmospheres. net (accessed May 18, 2010)*, 2008.

[315] Michael E Payne, Linh B Ngo, Flavio Villanustre, and Amy W Apon. Managing the academic data lifecycle: A case study of hpcc. In *IEEE International Conference on Big Data (BigData)*, pages 22–30. IEEE, 2014.

[316] Ralph Peeters and Christian Bizer. Entity matching using large language models. *arXiv preprint*, 2023.

[317] Jorge Pereira, Thais Batista, Everton Cavalcante, Arthur Souza, Frederico Lopes, and Nelio Cacho. A platform for integrating heterogeneous data and developing smart city applications. *Future Generation Computer Systems*, 128:552–566, 2022.

[318] Ricardo Lopes Pereira, Pedro Cruz Sousa, Ricardo Barata, André Oliveira, and Geert Monsieur. Citysdk tourism API - building value around open data. *J. Internet Services and Applications*, 6(1):24:1–24:13, 2015.

[319] Charith Perera, Yongrui Qin, Julio C. Estrella, Stephan Reiff-Marganiec, and Athanasios V. Vasilakos. Fog computing for sustainable smart cities: A survey. *ACM Comput. Surv.*, 50(3), June 2017.

[320] Riccardo Petrolo, Valeria Loscr´ı, and Nathalie Mitton. Towards a smart city based on cloud of things. In *International Workshop on Wireless and Mobile Technologies for Smart Cities (WiMobCity)*, pages 61–66. ACM, 2014.

[321] Judicaël Picaut, Nicolas Fortin, Erwan Bocher, Gwendall Petit, Pierre Aumond, and Gwenaël Guillaume. An open-science crowdsourcing approach for producing community noise maps using smartphones. *Building and Environment*, 148:20–33, 2019.

[322] Gabriele Picco, Marcos Martinez Galindo, Alberto Purpura, Leopold Fuchs, Vanessa Lopez, and Thanh Lam Hoang. Zshot: An open-source framework for zero-shot named entity recognition and relation extraction. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 357–368, July 2023.

[323] Gorkem Polat, Ilkay Ergenc, Haluk Tarik Kani, Yesim Ozen Alahdab, Ozlen Atug, and Alptekin Temizel. Class distance weighted cross-entropy loss for ulcerative colitis severity estimation. In *Medical Image Understanding and Analysis*, page 157–171. Springer-Verlag, 2022.

[324] Dan Pritchett. BASE: An ACID alternative. *ACM Queue*, 6(3):48–55, 2008.

[325] CITYkeys EU Horizon 2020 project.

[326] Achilleas Psyllidis. Ontology-based data integration from heterogeneous urban systems: A knowledge representation framework for smart cities. In *ACM*, 07 2015.

[327] Achilleas Psyllidis, Alessandro Bozzon, Stefano Bocconi, and Christiaan Bolivar. A platform for urban analytics and semantic data integration in city planning. In *Springer*, 07 2015.

[328] Dan Puiu, Payam Barnaghi, Ralf Tönjes, Daniel Kumper, Muhammad Intizar Ali, Alessandra Mileo, Josiane Parreira, Marten Fischer, Şefki Kolozali, Nazli Farajidavar, Feng Gao, Thorben Iggena, Thu-Le Pham, Cosmin-Septimiu Nechifor, Daniel Puschmann, and Joao Fernandes. Citypulse: Large scale data analytics framework for smart cities. *IEEE Access*, 4:1086–1108, 01 2016.

[329] Subashini Raghavan, Boung-Yew Lau Simon, Ying Loong Lee, Wei Lun Tan, and Keh Kim Kee. Data integration for smart cities: Opportunities and challenges. In *DataIF*, 2020.

[330] Shriram Tallam Puranam Raghu, Dawn T. MacIsaac, and Erik J. Scheme. Self-supervised learning via vicreg enables training of emg pattern recognition using continuous data with unclear labels. *Computers in Biology and Medicine*, 185:109479, 2025.

[331] Indra Raharjana, Daniel Siahaan, and Chastine Fatichah. User stories and natural language processing: A systematic literature review. *IEEE Access*, PP:1–1, 04 2021.

[332] Aryan Ratra, Aryan Agarwal, Satvik Vats, Vikrant Sharma, Vinay Kukreja, and Satya Prakash Yadav. A comprehensive review on crime patterns and trends analysis using machine learning. In *2023 Second International Conference on Augmented Intelligence and Sustainable Systems (ICAISS)*, pages 732–736. IEEE, 2023.

[333] Andrew Rebeiro-Hargrave, Pak Lun Fung, Samu Varjonen, Andres Huertas, Salla Sillanpää, Krista Luoma, Tareq Hussein, Tuukka Petäjä, Hilkka Timonen, Jukka Limo, Ville Nousiainen, and Sasu Tarkoma. City wide participatory sensing of air quality. *Frontiers in Environmental Science*, 9, 2021.

[334] Theodoros Rekatsinas, Manas Joglekar, Hector Garcia-Molina, Aditya Parameswaran, and Christopher Ré. Slimfast: Guaranteed results for data fusion and source reliability. In *International Conference on Management of Data (SIGMOD)*, page 1399–1414. ACM, 2017.

[335] RFSC. The reference framework for sustainable cities.

[336] Murilo B. Ribeiro and Kelly R. Braghetto. A scalable data integration architecture for smart cities: Implementation and evaluation. *Journal of Information and Data Management*, 2022.

[337] Diego O. Rodrigues, Azzedine Boukerche, Thiago H. Silva, Antonio A.F. Loureiro, and Leandro A. Villas. Smaframework: Urban data integration framework for mobility analysis in smart cities. In *Proceedings of the 20th ACM International Conference on Modelling, Analysis and Simulation of Wireless and Mobile Systems*, MSWiM 2017, pages 227–236. ACM, 2017.

[338] J. Rogstadius, M. Vukovic, C. A. Teixeira, V. Kostakos, E. Karapanos, and J. A. Laredo. Crisistracker: Crowdsourced social media curation for disaster awareness. *IBM Journal of Research and Development*, 57(5):4:1–4:13, Sep 2013.

[339] J. Roitsch and S. Watson. An overview of dyslexia: definition, characteristics, assessment, identification, and intervention. *Science Journal of Education*, 7:81–86, 2019.

[340] Morris Rosenberg. *Society and the Adolescent Self-Image*. Princeton University Press, 1965.

[341] Marit Rosol, Gwendolyn Blue, and Victoria Fast. Social justice in the digital age: re-thinking the smart city with nancy fraser. *UCCities - Global Urban Research at the University of Calgary Working Paper #1*, 2019.

[342] Franziska Rosser and John Balmes. Ozone and childhood respiratory health: A primer for us pediatric providers and a call for a more protective standard. *Pediatric Pulmonology*, 58(5):1355–1366, 2023.

[343] David Rubenstein, Wei Yin, and Mary D Frame. *Biofluid mechanics: an introduction to fluid mechanics, macrocirculation, and microcirculation*. Academic Press, 2015.

[344] Diego Sevilla Ruiz, Severino Feliciano Morales, and Jesús García Molina. Inferring Versioned Schemas from NoSQL Databases and Its Applications. In *International Conference on Conceptual Modeling (ER)*, pages 467–480, 2015.

[345] Raffaele Russo, Giuliano Di Giuseppe, Alessandro Vanacore, Valerio La Gatta, Antonino Ferraro, Antonio Galli, Marco Postiglione, and Vincenzo Moscato. Graph-based approach for european law classification. In *IEEE International Conference on Big Data (BigData)*, pages 1–9, 2023.

[346] Pramodkumar J. Sadalage and Martin J. Fowler. *NoSQL Distilled*. Addison-Wesley, 2012.

[347] Seref Sagiroglu and Duygu Sinanc. Big data: A review. In *2013 International Conference on Collaboration Technologies and Systems (CTS)*, pages 42–47. IEEE, 2013.

[348] Kosovare Sahatqija, Jaumin Ajdari, Xhemal Zenuni, Bujar Raufi, and Florije Ismaili. Comparison between relational and NOSQL databases. In *Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2018, Opatija, Croatia, May 21-25, 2018*, pages 216–221. IEEE, 2018.

[349] Tanvi Sahay, Ankita Mehta, and Shruti Jadon. Schema matching using machine learning. In *2020 7th International Conference on Signal Processing and Integrated Networks (SPIN)*, pages 359–366, 2020.

[350] Eduardo Felipe Zambom Santana, Ana Paula Chaves, Marco Aurelio Gerosa, Fabio Kon, and Dejan S Milojicic. Software platforms for smart cities: Concepts, requirements, challenges, and a unified reference architecture. *ACM Computing Surveys (CSUR)*, 50(6):78, 2018.

[351] Pedro M. Santos, João G. P. Rodrigues, Susana B. Cruz, Tiago Lourenço, Pedro M. d'Orey, Yunior Luis, Cecília Rocha, Sofia Sousa, Sérgio Crisóstomo, Cristina Queirós, Susana Sargento, Ana Aguiar, and João Barros. Portolivinglab: An iot-based sensing platform for smart cities. *IEEE Internet of Things Journal*, 5(2):523–532, 2018.

[352] Parthasarathy Saravanan, Jeganathan Selvaprabu, L Arun Raj, A Abdul Azeez Khan, and K Javubar Sathick. Survey on crime analysis and prediction using data mining and machine learning techniques. In *Advances in Smart Grid Technology: Select Proceedings of PECCON 2019—Volume II*, pages 435–448. Springer, 2021.

[353] SAREF. Saref semantic model for smart cities.

[354] F. Schummer and M. Hyba. An approach for system analysis with model-based systems engineering and graph data engineering. *Data-Centric Engineering*, 3:e33, 2022.

[355] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128(2):336–359, October 2019.

[356] Nimra Shahid, Munam Ali Shah, Abid Khan, Carsten Maple, and Gwanggil Jeon. Towards greener smart cities and road traffic forecasting using air pollution data. *Sustainable Cities and Society*, 72:103062, 2021.

[357] A. Shakirov, A. Molchanov, L. Ismailova, and M. Mezghani. Quantitative assessment of rock lithology from gamma-ray and mud logging data. *Geoenergy Science and Engineering*, 225:211664, 2023.

[358] Ankita Sharma, Xuanmao Li, Hong Guan, Guoxin Sun, Liang Zhang, Lanjun Wang, Kesheng Wu, Lei Cao, Erkang Zhu, Alexander Sim, Teresa Wu, and Jia Zou. Automatic data transformation using large language model - an experimental study on building energy data. In *IEEE International Conference on Big Data (BigData)*, pages 1824–1834, 2023.

[359] Y. Shavit, S. Agarwal, M. Brundage, S. Adler, C. O'Keefe, R. Campbell, and D. G. Robinson. Practices for governing agentic ai systems. *Research Paper, OpenAI*, 2023.

[360] Sally Shaywitz and Bennett Shaywitz. Dyslexia (specific reading disability). *Biological psychiatry*, 57(11):1301–1309, 2005.

[361] Jeff Shute et al. F1: A distributed SQL database that scales. *PVLDB*, 6(11):1068–1079, 2013.

[362] Jorge Silva, João Gabriel Almeida, Thaís Batista, and Everton Cavalcante. Aquedücte: A data integration service for smart cities. In Adriano César Machado Pereira and Leonardo Chaves Dutra da Rocha, editors, *WebMedia 2021: Brazilian Symposium on Multimedia and the Web, Belo Horizonte, Minas Gerais, Brazil, November 5-12, 2021*, pages 177–180. ACM, 2021.

[363] Aditi Singh, Abul Ehtesham, Saket Kumar, and Tala Talaei Khoei. Agentic retrieval-augmented generation: A survey on agentic rag, 2025.

[364] Rajesh Kumar Singh, H.R. Murty, S.K. Gupta, and A.K. Dikshit. An overview of sustainability assessment methodologies. *Ecological Indicators*, 15(1):281 – 299, 2012.

[365] Khanin Sisaengsuwanchai, Navapat Nananukul, and Mayank Kejriwal. How does prompt engineering affect chatgpt performance on unsupervised entity resolution? *arXiv preprint*, 2023.

[366] U. Sivarajah, M. M. Kamal, Z. Irani, and V. Weerakkody. Critical analysis of big data challenges and analytical methods. *Journal of Business Research*, 70(1):263–286, 2017.

[367] Leslie N. Smith. A disciplined approach to neural network hyper-parameters: Part 1 – learning rate, batch size, momentum, and weight decay, 2018.

[368] Ola Söderström, Till Paasche, and Francisco Klauser. Smart cities as corporate storytelling. *City*, 18(3):307–320, 2014.

[369] Adir Solomon, Mor Kertis, Bracha Shapira, and Lior Rokach. A deep learning framework for predicting burglaries based on multiple contextual factors. *Expert Systems with Applications*, 199:117042, 2022.

[370] Mehdi Sookhak, Helen Tang, Ying He, and F. Richard Yu. Security and privacy of smart cities: A survey, research issues and challenges. *IEEE Communications Surveys & Tutorials*, 21(2):1718–1743, 2019.

[371] Stephan Spiegel, Julia Gaebler, Andreas Lommatzsch, Ernesto De Luca, and Sahin Albayrak. Pattern recognition and classification for multivariate time series. In *International Workshop on Knowledge Discovery from Sensor Data*, page 34–42. ACM, 2011.

[372] Telecommunication standardization sector of ITU (ITU-T). Kpis on smart sustainable cities.

[373] Telecommunication standardization sector of ITU (ITU-T). Smart sustainable cities: An analysis of definitions. *Focus Group Technical Report*, 2014.

[374] Telecommunication standardization sector of ITU (ITU-T). Itu-t y.4400 series - smart sustainable cities - setting the framework for an ict architecture. *ITU-T Y-series Recommendations*, Supplement 27, 2016.

[375] Telecommunication standardization sector of ITU (ITU-T). Key performance indicators for smart sustainable cities to assess the achievement of sustainable development goals. *ITU-T Recommendation*, Y.4903/L.1603, 2016.

[376] Telecommunication standardization sector of ITU (ITU-T). Key performance indicators related to the use of information and communication technology in smart sustainable cities. *ITU-T Recommendation*, Y.4901/L.1601, 2016.

[377] Telecommunication standardization sector of ITU (ITU-T). Overview of key performance indicators in smart sustainable cities. *ITU-T Recommendation*, Y.4900/L.1600, 2016.

[378] Telecommunication standardization sector of ITU (ITU-T). Overview of key performance indicators in smart sustainable cities. *ITU-T Recommendation*, Y.4902/L.1602, 2016.

[379] Telecommunication standardization sector of ITU (ITU-T). Smart sustainable cities maturity model. *ITU-T Recommendation*, Y.4904, 2019.

[380] Telecommunication standardization sector of ITU (ITU-T). Key performance indicators for smart sustainable cities to assess the achievement of sustainable development goals. *ITU-T Recommendation*, Y.4903, 2022.

[381] Michael Stonebraker, Uundefinedur Çetintemel, and Stan Zdonik. The 8 requirements of real-time stream processing. *International Conference on Management of Data (SIGMOD)*, 34(4):42–47, dec 2005.

[382] Veda C. Storey, Roman Lukyanenko, and Arturo Castellanos. Conceptual modeling: Topics, themes, and technology trends. *ACM Comput. Surv.*, 55(14s), jul 2023.

[383] Xiang Su, Jukka Riekki, Jukka K. Nurminen, Johanna Nieminen, and Markus Koskimies. Adding semantics to internet of things. *Concurrency and Computation: Practice and Experience*, 27(8):1844–1860, 2015.

[384] Xiang Su, Hao Zhang, Jukka Riekki, Ari Keränen, Jukka K. Nurminen, and Libin Du. Connecting iot sensors to knowledge-based systems by transforming senml to rdf. *Procedia Computer Science*, 32:215–222, 2014. The 5th International Conference on Ambient Systems, Networks and Technologies (ANT-2014), the 4th International Conference on Sustainable Energy Information Technology (SEIT-2014).

[385] Qi Sun, Kun Huang, Xiaocui Yang, Rong Tong, Kun Zhang, and Soujanya Poria. Consistency guided knowledge retrieval and denoising in llms for zero-shot document-level relation triplet extraction. *arXiv preprint*, 2024.

[386] Iryna Susha, Marijn Janssen, and Stefaan Verhulst. Data collaboratives as "bazaars"? a review of coordination problems and mechanisms to match demand for data with supply. *Transforming Government: People, Process and Policy*, 11(1):157–172, 2017.

[387] Yusuke Takamori, Junya Sato, Masahiro Fujimoto, Masaki Endo, Shigeyoshi Ohno, Daiju Kato, and Hiroshi Ishikawa. Current status of examples of initiatives using open data in government. *Government*, 2021.

[388] Ruixiang Tang, Xiaotian Han, Xiaoqian Jiang, and Xia Hu. Does synthetic data generation of llms help clinical text mining? *arXiv preprint*, 2023.

[389] Yachen Tang, Xingping Wu, Chunlei Zhou, Guangxin Zhu, Jinwei Song, Guangyi Liu, and Zhihong Li. Automatic schema construction of electrical graph data platform based on multi-source relational data models. *Data & Knowledge Engineering (DKE)*, 145:761–765, 2023.

[390] Toby J. Teorey and James P. Fry. The logical record access approach to database design. *ACM Comput. Surv.*, 12(2):179–211, 1980.

[391] Tokyo. Tokyo metropolitan government open data catalogue (translated from japanese).

[392] Martin Tomitsch, Joel Fredericks, Dan Vo, Jessica Frawley, and Marcus Foth. Non-human personas: Including nature in the participatory design of smart cities. *Interaction Design and Architecture(s)*, 50(50):102–130, 2021.

[393] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023.

[394] Sandhya Tripathi, Bradley Fritz, Mohamed Abdelhack, Michael Avidan, Yixin Chen, and Christopher King. Deep learning to jointly schema match, impute, and transform databases. *A*, 06 2022.

[395] Juan Trujillo, Karen Davis, Xiaoyang Du, Ernesto Damiani, and Veda Storey. Conceptual modeling in the era of big data and artificial intelligence: Research topics and introduction to the special issue. *Data & Knowledge Engineering (DKE)*, 135, 2021.

[396] Immanuel Trummer. DB-BERT: A database tuning tool that "reads the manual". In *International Conference on Management of Data (SIGMOD)*, page 190–203. ACM, 2022.

[397] Immanuel Trummer. From bert to gpt-3 codex: harnessing the potential of very large language models for data management. *Proc. VLDB Endow.*, 15(12):3770–3773, aug 2022.

[398] Thuy Truong, Ahmed Khalid, and Philip Leroux. Optimized version of reference architecture including update to iot interfaces. cutler d2.5. *International journal of information management*, 2020.

[399] Filareti Tsalakanidou, Ekaterina Gilman, Panos Kostakos, and Andrew Byrne. Requirements for data crawling, integration and anonymization. cutler d3.1. *CUTLER*, 2018.

[400] Mio Tsubakimoto. Current status and issues of university education-related data in tokyo open data. *IIAI Letters on Institutional Research*, 1, 2022.

[401] Uber. Driving solutions to build smarter cities, 2015.

[402] European Union. Cities of tomorrow. challenges, visions, ways forward, 2011.

[403] Matthias Urban, Duc Dat Nguyen, and Carsten Binnig. Omniscientdb: A large language model-augmented dbms that knows what other dbmss do not know. In *International Workshop on Exploiting Artificial Intelligence Techniques for Data Management (aiDM)*. ACM, 2023.

[404] Muhammad Usman, Mian Ahmad Jan, Xiangjian He, and Jinjun Chen. A survey on big multimedia data processing and management in smart cities. *ACM Comput. Surv.*, 52(3), June 2019.

[405] Maarten van Steen and Andrew S. Tanenbaum. *Distributed Systems*. Pearson, 2017.

[406] Polychronis Velentzas, Antonio Corral, and Michael Vassilakopoulos. *Big Spatial and Spatio-Temporal Data Analytics Systems*, pages 155–180. Springer Berlin Heidelberg, Berlin, Heidelberg, 2021.

[407] Jayant Venkatanathan, Denzil Ferreira, Michael Benisch, Jialiu Lin, Evangelos Karapanos, Vassilis Kostakos, Norman Sadeh, and Eran Toch. Improving users' consistency when recalling location sharing preferences. In *Human-Computer Interaction (INTERACT)*, pages 380–387. Springer, 2011.

[408] Vaughn Vernon. *Implementing Domain-Driven Design*. Addison-Wesley, 2013.

[409] Vertx. Vert.x.

[410] Jenni Viitanen and Richard Kingston. Smart cities and green growth: Outsourcing democratic and environmental resilience to the global technology sector. *Environment and Planning A: Economy and Space*, 46(4):803–819, 2014.

[411] Félix J. Villanueva, Maria J. Santofimia, David Villa, Jesús Barba, and Juan Carlos López. Civitas: The smart city middleware, from sensors to big data. In *2013 Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, pages 445–450, 2013.

[412] Massimo Villari, Maria Fazio, Schahram Dustdar, Omer Rana, Devki Nandan Jha, and Rajiv Ranjan. Osmosis: The osmotic computing platform for microelements in the cloud, edge, and internet of things. *Computer*, 52(8):14–26, 2019.

[413] Aku Visuri, Zeyun Zhu, Denzil Ferreira, Shinichi Konomi, and Vassilis Kostakos. Smartphone detection of collapsed buildings during earthquakes. In *ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers*, UbiComp 2017, page 557–562. ACM, 2017.

[414] Deepak Vohra. Apache parquet. In *Practical Hadoop Ecosystem*, pages 325–335. Springer, 2016.

[415] VoltDB. Voltdb.

[416] Vrailexia. VRAILEXIA - Into The Box.

[417] Vrailexia. Vrailexia home page.

[418] Shiraz Ali Wagan, Muhammad Junaid, Nawab Muhammad Faseeh Qureshi, Dong Ryeol Shin, and Keehyun Choi. Comparative survey on

big data security applications, a blink on interactive security mechanism in apache ozone. In *2020 Global Conference on Wireless and Optical Technologies (GCWOT)*, pages 1–6. IEEE, 2020.

[419] Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, and Guoyin Wang. Gpt-ner: Named entity recognition via large language models. *arXiv preprint*, 2023.

[420] Xuelin Wang and Qihao Yang. LingX at ROCLING 2023 MultiNER-health task: Intelligent capture of Chinese medical named entities by LLMs. In *Conference on Computational Linguistics and Speech Processing*, pages 350–358. ACLCLP, October 2023.

[421] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Huai hsin Chi, F. Xia, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint*, abs/2201.11903, 2022.

[422] Ben Williamson. Educating the smart city: Schooling smart citizens through computational urbanism. *Big Data & Society*, 2(2):2053951715617783, 2015.

[423] John Winn, John Guiver, Sam Webster, Yordan Zaykov, Martin Kukla, and Dany Fabian. Alexandria: Unsupervised high-precision knowledge base construction using a probabilistic program. In *AKBC*, 2018.

[424] Di Wu, Weite Feng, Tong Li, and Zhen Yang. Evaluating the intelligence capability of smart homes: A conceptual modeling approach. *Data & Knowledge Engineering (DKE)*, 148:102218, 2023.

[425] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint*, 2016.

[426] Haoran Xu, Yunmo Sharaf, Amr Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. Contrastive Preference Optimization: Pushing the Boundaries of LLM Performance in Machine Translation. *arXiv preprint*, 2024.

[427] Jialu Yan. Research on data analysis application based on spark computing. *European Journal of AI, Computing & Informatics*, 1(2):23–29, Jun. 2025.

[428] Chao-Lung Yang, Zhi-Xuan Chen, and Chen-Yi Yang. Sensor classification using convolutional neural network by encoding multivariate time series as two-dimensional colored images. *Sensors*, 20:168, 12 2019.

[429] Tan Yigitcanlar. Smart city beyond efficiency: Technology-policy-community at play for sustainable urban futures. *Housing Policy Debate*, 31(1):88–92, 2021.

[430] Tan Yigitcanlar, Marcus Foth, and Md. Kamruzzaman. Towards post-anthropocentric cities: Reconceptualizing smart cities to evade urban ecocide. *Journal of Urban Technology*, 26(2):147–152, 2019.

[431] Johanna Ylipulli and Aale Luusua. Without libraries what have we? public libraries as nodes for technological empowerment in the era of smart cities, ai and big data. In *International Conference on Communities & Technologies - Transforming Communities*, pages 92–101. ACM, 2019.

[432] Moayid Ali Zaidi. Conceptual modeling interacts with machine learning - A systematic literature review. In *Computational Science and Its Applications (ICCSA)*, volume 12957, pages 522–532. Springer, 2021.

[433] Zensors. Zensors: Smart Video Analytics.

[434] Hui Zhang, Ju Ji, Guangchen Ruan, Neel Patel, Regan Giesting, Leah Miller, Yi Lin Yang, and Jian Yang. Digital data platform for connected clinical trials. In *2022 IEEE International Conference on Big Data (Big Data)*, pages 2446–2453, 2022.

[435] R Zhang, H Du, Y Liu, D Niyato, J Kang, S Sun, and H V Poor. Interactive ai with retrieval-augmented generation for next generation networking. *IEEE Network*, 2024.

[436] Xinwei Zhao, Saurabh Garg, Carlos Queiroz, and Rajkumar Buyya. Chapter 11 - a taxonomy and survey of stream processing systems. In Ivan Mistrik, Rami Bahsoon, Nour Ali, Maritta Heisel, and Bruce Maxim, editors, *Software Architecture for Big Data and the Cloud*, pages 183–206. Morgan Kaufmann, Boston, 2017.

[437] Yu Zheng. *Urban computing*. The MIT Press, 2018.

[438] Yu Zheng, Licia Capra, Ouri Wolfson, and Hai Yang. Urban computing: Concepts, methodologies, and applications. *ACM Trans. Intell. Syst. Technol.*, 5(3):38:1–38:55, September 2014.

[439] Qing Zhu, Fan Zhang, Shan Liu, and Yuze Li. An anticrime information support system design: Application of k-means-vmd-bigru in the city of chicago. *Information & Management*, 59(5):103247, 2022.

[440] Rui Zhu, Man Sing Wong, Mei-Po Kwan, Min Chen, Paolo Santi, and Carlo Ratti. An economically feasible optimization of photovoltaic provision using real electricity demand: A case study in new york city. *Sustainable Cities and Society*, 78:103614, 2022.

[441] Aman Zhumekeshov, Mukhtar Kunyrbayev, Ilyas Turgazinov, Daniyar Nassipov, Shukhrat Mametov, and Kassymzhomart Ulasbek. Automating conventional well correlation using machine learning techniques, 2023.

[442] Esteban Zimányi, Mahmoud Sakr, and Arthur Lesuisse. Mobilitydb: A mobility database based on postgresql and postgis. *ACM Trans. Database Syst.*, 45(4), dec 2020.

[443] Sotiris Zygiaris. Smart city reference model: Assisting planners to conceptualize the building of smart city innovation ecosystems. *Journal of the Knowledge Economy*, 4(2):217–231, 2013.

Glossary

2D CNN-LSTM Two-Dimensional Convolutional Neural Network-Long Short-Term Memory (2D CNN-LSTM).

AWS Amazon Web Services (AWS).

BPM Beam Position Monitors (BPM) are the non-destructive diagnostics used most frequently at nearly all linacs, cyclotrons, and synchrotrons. BPMs deliver the centre of mass of the beam and act as a monitor for the longitudinal bunch shape.

BSON BSON (Binary JSON), a variant of the popular JSON format.

CoT Cloud of Things (CoT) refers to integration of Internet of Things (IoT) with Cloud Computing (CC)

DL Deep Learning (DL),is a subfield of machine learning (ML) that utilizes artificial neural networks with multiple layers to analyze data and make intelligent decisions.

DTW Dynamic Time Warping (DTW) is an algorithm for measuring similarity between two temporal sequences.

eCRF The “Cahier d’observation électronique (eCRF), is a numeric booklet of data about the patiente-CRF.

EFTA European Free Trade Association (EFTA) is an intergovernmental organization focused on promoting free trade and economic integration among its member countries.

ETSI The European Telecommunications Standards Institute (ETSI) is an independent, not-for-profit, standardization organization operating in the field of information and communications.

FCC The Future Circular Collider Study (FCC) is developing designs for a new research infrastructure to host the next generation of higher performance particle colliders.

FCNs Fully Convolutional Networks (FCNs), are an architecture used mainly for semantic segmentation.

FCS	Flow Cytometry Standard (FCS) is a data file standard for the reading and writing of data from flow cytometry experiments.
GADF	Gramian Angular Difference Field (GADF)
GARDD	GrAph Schema foR Dyslexic Disorders (GARDD), a conceptual schema for representing dyslexic disorders.
GeoTS	A Time Series Classification framework for estimating geological formation to model carbon storage reservoirs (GeoTS).
GR	Gamma Rays (GR) logs measure radioactivity to determine what types of rocks are present in the well.
Grad-CAM	Gradient-weighted Class Activation Mapping (Grad-CAM), is a technique used to visualize which parts of an image a Convolutional Neural Network (CNN) focuses on when making a classification decision.
HDBSCAN	HDBSCAN is a density-based clustering algorithm that constructs a cluster hierarchy tree and then uses a specific stability measure to extract flat clusters from the tree.
HDFS	Hadoop Distributed File System (HDFS) is a core component of the Apache Hadoop framework and implements a distributed file system designed to store large data sets across multiple commodity hardware.
HPCC	High-Performance Computing Cluster (HPCC), also known as Data Analytics Supercomputer (DAS), is an open source, data-intensive computing system platform developed by LexisNexis Risk Solutions.
ICT	Information and Communication Technology (ICT), refers to all technologies used to handle information and aid communication, encompassing both computer and network hardware, as well as their software.
IoT	Internet of Things (IoT) refers to the network of physical objects embedded with sensors, software, and other technologies that allow them to connect and exchange data with other devices and systems over the internet.

ISO International Organization for Standardization (ISO), is a non-governmental international organization that develops and publishes voluntary international standards.

ITU International Telecommunication Union (ITU) is a specialized agency of the United Nations that focuses on information and communication technologies (ICTs).

KMS Key Management Server (KMS) is an Hadoop competent that provides cryptographic key management server based on Hadoop.

KPI Key Performance Indicator (KPI) is a quantifiable measure of performance over time for a specific objective.

LLM Large Language Model (LLM) is a language model trained with self-supervised machine learning on a vast amount of text, designed for natural language processing tasks.

LSTM Long Short-Term Memory (LSTM) is a type of recurrent neural network (RNN) aimed at mitigating the vanishing gradient problem commonly encountered by traditional RNNs.

LSTM-FCN Long Short Term Memory Fully Convolutional Network (LSTM-FCN).

LSTM-XMC Long Short-Term Memory (LSTM) is the augmentation of LSTM with XCM submodules. We combine an LSTM parallel network with the existing 1D and 2D parallel networks of the XCM.

MAE Mean Absolute Error (MAE) is a metric used to evaluate the performance of regression models in machine learning.

MAE Time Series Classification (TSC) is a machine learning technique used to categorize time-ordered data into predefined classes.

NER Named Entity Recognition (NER) is a natural language processing (NLP) method that extracts information from text.

NGSI-LD NGSI-LD is an information model and API for publishing, querying and subscribing to context information.

NLP Natural Language Processing (NLP) is a field of Artificial Intelligence that focuses on enabling computers to understand, interpret, and generate human language.

NOAM NoSQL Abstract Model (NoAM) is a high-level data model for NoSQL databases.

PROCLAIM PROCLAIM (PROfile-based Cluster-Labeling for AttrIbute Matching) a metamodel that performs an automatic, unsupervised clustering-based approach to match attributes of a large number of heterogeneous sources.

U4SSC Smart Sustainable Cities (U4SSC) is a global initiative that also provides an international platform for information exchange and partnership building to guide cities and communities in achieving the United Nations Sustainable Development Goals.

XCM eXplainable Convolutional neural network (XCM) is acompact convolutional neural network which extracts information relative to the observed variables and time directly from the input data.