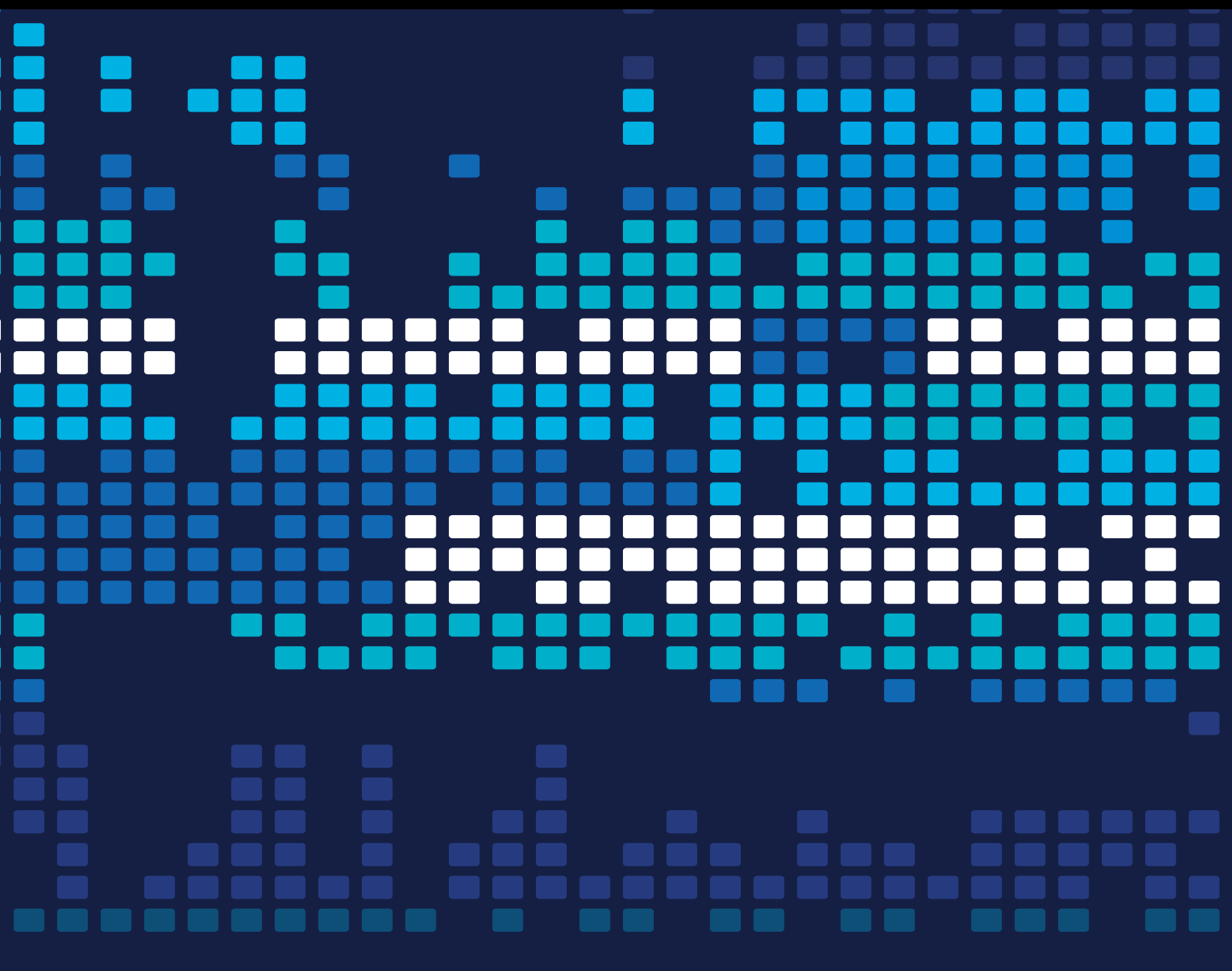
Scientific Programming

Data Driven Computational Intelligence for Scientific Programming

Lead Guest Editor: Alvaro Rubio-Largo
Guest Editors: Juan Carlos Preciado and Luis Iribarne



Data Driven Computational Intelligence for Scientific Programming

Scientific Programming

Data Driven Computational Intelligence for Scientific Programming

Lead Guest Editor: Alvaro Rubio-Largo
Guest Editors: Juan Carlos Preciado and Luis Iribarne

Editorial Board

Contents

Hindawi
Scientific Programming
Volume 2019, Article ID 5235706, 4 pages
https://doi.org/10.1155/2019/5235706

Editorial

Data-Driven Computational Intelligence for Scientific Programming

Álvaro Rubio-Largo,[1] **Juan Carlos Preciado**,[2] **and Luis Iribarne**[3]

[1]*Universidade Nova de Lisboa, Lisboa 1099-085, Portugal*
[2]*University of Extremadura, Cáceres 10003, Spain*
[3]*University of Almería, Almería 04120, Spain*

Correspondence should be addressed to Álvaro Rubio-Largo; arl@unex.es

Received 22 July 2019; Accepted 22 July 2019; Published 19 August 2019

In recent years, big data and its potential to shed valued insights into enhanced decision-making processes has attracted an increasing interest from both academia and industry. Throughout history, there have been eras that have marked turning points in society. Currently, we are faced with the beginning of one of these turning points, a new leap in evolution which arises thanks to the advantages of technology, but which has recently been revolutionizing technical concepts of development and programming. We are in a new age, the era of Data [1].

Currently, the volume of data generated each day is very high, coming from different multiple sources and in diverse formats, usually designed with different goals, methods, profiles, and production and consumption rhythms. The amount of data generated by businesses, public administrations, and numerous industrial and scientific research facilities has increased immeasurably in the past years, turning traditional systems into complex or supercomplex systems. These data may be structured, semistructured, and/or unstructured, extracted from sources as different as Natural Language Processing [2] (chatbots, comments, and social media), multimedia content (videos, images, and audio), geographic information systems (GIS), or sensors (Internet of Things/Everything) on a wide variety of platforms or environments (e.g., machine-to-machine communications, social media sites, sensors networks, or natural interaction) [3].

These are highly relevant data, so much so that they must be taken into account when designing and developing the solutions of the future (or redesigning the present ones). The proper evolution of data concept is of vital importance, due to the fact that they have a huge impact on the economic and social development of the institutions and enterprises they belong to.

In this new cycle of massive data, our administrations, enterprises, and citizens generate colossal amounts of data, data which on their own do not help us in our everyday life or in making decisions, and hence the importance of treating these data in order to turn them into information that is useful and relevant for them [4].

The evolution of computational intelligence and big data practices stimulates an increasing interest in Analytics solution [5]. Accurate prediction, precise identification of trends, and discovering behaviour patterns can optimize the resource usage or consumption, generating new knowledge in science and research, and enabling faster and better decisions in politics, weather, science, research, real estate, sports, or healthcare, among other social application domains [2].

To date, the solutions proposed have been disjointed, and actions put into practice as a result of fashions or through the implantation of a certain known technology, but without a perspective of the specific need of the concept of amount of data. The challenge is no longer the evolution and development of the technology, the current issue is not related with the production and acquisition of data, and now we are treating with new challenges and problems regarding the processing of these data to build information [6]. Based on these concepts and the engineering of complex massive data systems with high availability, we see

that a computational intelligence environment is configured like a system, a smart system. In this context, methodology supporting the design of this system would firstly ask the computational intelligence system's designer, to identify the requirements for the existing data required to continue with the other phases.

The great data acquisition obliges us to have innovative massive data storage systems on a scale not yet contemplated. This mass of data is growing exponentially, and 90% of data on a worldwide scale has been created only in the last two years [7]. The management and processing of this data defines a new knowledge management paradigm which can only be addressed by means of constant methodological and technological innovation regarding the field of computational intelligence applied to scientific programming [8].

The different data sources and formats result in a complexity which makes decision-making expensive, and in cases where the implementation of Big Data, Business Intelligence, and Business Analytics is not enough, users need more and better ways of understanding data. In this scenario, techniques like Data Visualization come into the scene, which enable the visual simplification of all information built from the collected data. This information visualization supports decision-making and the advanced analysis of the data collected, by means of report and graphical representation techniques, with data visualization capacities which enable actors to build the necessary information flexibly on the same set of data, based on this data and placing it or other data at the service of third parties, hence providing an environment of rich collaboration and innovation. One of the trends given clear importance by Computation Intelligence Systems is the visualization of information in a simple, agile, and powerful way, i.e., Computation Intelligence Systems makes possible the process data and display information quickly and in a reasonable time frame to assimilate all possible information by the people for whom it is intended. Definitively, the advance in technology and scientific programming we are experiencing plays an essential role in the evolution of computational intelligence environment [9].

Computational intelligence techniques form a set of nature-inspired computational methodologies and techniques which have been developed to face the aforementioned complex scientific programming, for which traditional models are unable to work due to high complexity, uncertainty, and the stochastic nature of processes. These techniques typically include parallel/distributed pattern-recognition techniques, genetic programming, fuzzy systems, or evolutionary computation. The overall aim of this special issue is to collect state-of-the-art research findings on the latest developments, as well as up-to-date issues and challenges in the field of computational intelligence applied to scientific programming.

This special issue is configured like a collection of papers on a hot topic of increasing interest within of scientific programming, presenting and describing new research or applications in this field. Several call for papers were sent and distributed among the main mailing lists of the field for relevant researchers to submit their research to this special issue. As an example of the current interest in this field, it is worth mentioning that, for this special issue, we have managed a total of 15 high-quality submissions from different countries: Spain, China, India, Greece, Australia, United Kingdom, and Vietnam. These researches have been managed according to the terms and guidelines of this journal. All the papers included in this special issue were reviewed by at least two expert reviewers. Furthermore, all the papers in the special issue received a minimum of two review rounds. Finally, five papers of high quality in emerging research areas were accepted for inclusion in the special issue (acceptance rate = 5/15 = 33.33%). In summary, we think these papers bring us an international sampling of significant work.

In general, the papers included in this special issue cover detailed scientific aspects related mainly with the fields of recommendation systems, machine learning, pattern recognition, spatial data interaction, and complex events. Research advances are applied to different social domains such as smart cities, scientific digital libraries, urban spatial data, and public health.

The title of our first paper is "Study of Urban System Spatial Interaction Based on Microblog Data: A Case of Huaihe River Basin, China," by Y. Fan, J. Yao, Z. He, B. He, and M. Li. This paper obtains the user microblog information through the sina microblog open platform and studies the urban spatial pattern and urban interaction by means of statistical analysis and spatial analysis. This paper takes the Huaihe basin as the case area to verify the proposal presented on it.

The main research conclusions from our first paper are as follows: (1) the data interface provided by the microblog platform can study the urban spatial pattern. The user trajectory of microblog data can explore the spatial relationship of regional cities, and data acquisition and data quality evaluation can meet the research requirements, and (2) based on microblog data, the spatial and temporal characteristics of the urban system spatial pattern in the Huaihe River Basin are analyzed from network connectivity and urban interaction. The study found that the urban spatial relation in the Huaihe River Basin has the following characteristics: the spatial difference of urban size distribution is obvious; urban layout presents a stratified aggregation phenomenon; and the high-grade cities lead the city's interaction.

As for the application of microblog data in urban research, the current data mainly focus on information text, social relations, and other aspects. The research is mainly about event detection and hot spot exploration. The combination of big data thinking and data mining technology will have more research findings in the study of urban problems.

The second paper is "Practical Experiences in the Use of Pattern-Recognition Strategies to Transform Software Project Plans into Software Business Processes of Information Technology Companies" by C. Arevalo, I. Ramos, J. Gutiérrez, and M. Cruz. The authors' proposal provides a framework to generate software business processes that would otherwise be hidden or wasted in

databases of non-process-aware information systems (non-PAISs). This hidden knowledge can be used to implement the business process management approach in information technology companies (ITCs) that will help them to become more competitive and reduce costs. Compared to other business process discovery methods used with non-PAISs, their results are more adjusted to the reality of processes since they focus on transformations among artifacts that are close to executed processes that exist at different levels of abstraction (i.e., platform level and software expert level). Furthermore, business processes may be enriched with data regarding resources and costs that may also be bound to projects in project management systems. This way, new data will be available to set metrics and study key performance indicators of software business processes.

This paper illustrates the AQUA-WS project case study to test the developed MDA-based roadmap. In this case study, they have shown that generated processes are similar to real processes that a business software expert may design. For this reason, they have delivered a semiautomatic proposal to obtain processes of ITCs. As future work, the authors say that they will be able to use further source systems, such as other PMSs, ECMs, ERPs, CRMs, SCMs, or tailor-made software. Besides, they will propose roadmaps to specific SPMLs or GPMLs targets, used by ITCs that work with this type of systems. In this case study, they have considered the following aspects of source systems, target systems, and heuristics to generate business processes.

Our third paper is "A High-Frequency Data-Driven Machine Learning Approach for Demand Forecasting in Smart Cities" by J. C. Preciado, A. E. Prieto, R. Benitez, R. Rodríguez-Echeverría, and J. M. Conejero. This paper presents an approach based on pattern-similarity techniques to forecast water demand, referred to as short-term pattern similarity (STPS). This work faces two important challenges that have been traditionally neglected in previous approaches, namely, a high frequency of predictions (based on measurements in terms of minutes) and the need for external data such as annual seasonality or weather that increases the complexity of the approaches. In that sense, on the one hand, it is based on 1 min steps predictions, and, on the other hand, it does not require estimating annual seasonality since it determines this seasonality by constructing the X and Y patterns in which the series has been normalized.

In order to validate their approach, the study was applied over three different sites of a city in northern Spain. The results obtained provided interesting insights, such as the best predictions obtained in high-density population areas, the difficulties for identifying patterns for Sundays in industrial areas, or the higher random behaviour in low-density areas. Additionally, their pattern-similarity approach (STPS) was also compared to other similar techniques that have been previously used for water forecasting, i.e., $\alpha\beta$ water demand forecast ($\alpha\beta$-WDF) and generalized regression neural network (GRNN). The results obtained evidenced that $\alpha\beta$-WDF was the approach with worst results whilst GRNN and STPS behave similarly. As future work, the authors try to manage some weaknesses already identified in the proposed method. Firstly, predictions success is lower when anomalous days are taken into account. Secondly, with the aim of improving the results for data sites where apparently there is not regularity, such as the low-density population area in our study, other approach like the shorter prediction horizons could be considered, for instance, 4–6 hours. Notwithstanding, this is a subject that remains currently untested. Finally, another interesting line of further work is the application of the presented method for water distribution in different cities with similar water requirements.

A systematic mapping study is addressed in the fourth paper, "Recommendation and Classification Systems: A Systematic Mapping Study," by J. G. Enríquez, L. Morales-Trujillo, F. Calle-Alonso, F. J. Domínguez-Mayo, and J. M. Lucas-Rodríguez. This study has been performed to facilitate researchers and practitioners the task of choosing the most appropriate system, technology, or algorithm to include in the ADAGIO project for satisfying their requirements. In this sense, this paper presents a systematic mapping study (SMS) that analyzes the current state of the art of the recommendation and classification systems and how they work together. Then, from the point of view of the software development life cycle, this review also shows that the work being done in the ML (for classification and recommendation) research and industrial environment is far from earlier stages such as business requirements and analysis. This makes it very difficult to find efficient and effective solutions that support real business needs from an early stage. Then, this paper suggests the development of new ML research lines to facilitate its application in the different domains. As future work, the authors propose a very interesting research line may focus on how to combine these systems to obtain more efficient and effective solutions.

Unlike most SMSs that are focused on the scientific literature, this study has been carried out from two points of view as discussed throughout the paper: the scientific and the industrial scopes. Within the scientific field, the results showed that the most studied technique in recommendation systems is recommendation with the use of collaborative filters, closely followed by those that use content-based filters. Only 14 used hybrid recommendation systems, whereas 31 used collaborative filtering and 29 used content-based methods. This is an interesting suggestion for researchers starting to use recommender systems, to find which of them are more popular and more used in the scientific environment. By conducting market research through systematic industrial mapping, it was found that there are many technologies that offer automatic learning solutions, and most of which are complete systems or libraries. However, the nature of most of them could not be known because the proprietary software did not allow it. Another important issue that must be highlighted is that not only the communities of free software developers are interested in this topic but also there are large companies that are working on it

for commercial purposes. This clearly shows the underlying economic interest, an indicator that it is a branch of long-distance research.

The fifth paper is entitled "Facilitating the Quantitative Analysis of Complex Events through a Computational Intelligence Model-Driven Tool" by G. Díaz, H. Maciá, V. Valero, J. Boubeta-Puig, and G. Ortiz. Complex event processing (CEP) is a computational intelligence technology capable of analyzing big data streams for event pattern recognition in real time. In this paper, the authors illustrate the use of the MEdit4CEP–CPN approach for the complex event analysis through a case study based on the sick building syndrome. The event patterns have been graphically modeled with MEdit4CEP–CPN and then automatically transformed into both Event Processing Languages (EPL) and Coloured Petri Nets (CPN) code. Additionally, CPN Tools has been used to make quantitative analysis of events produced for this case study. Given the flexibility provided by MEdit4CEP–CPN, this analysis could be applied to other cutting-edge real-world case studies, such as eHealth, robotic, and mobile edge, and cloud computing applications.

The main advantage of MEdit4CEP-CPN is that supports many functionalities that other approaches do not provide, such as (1) modeling CEP domains and event patterns in a user-friendly way by dragging and dropping elements on a canvas, (2) validating the pattern syntax, (3) automatically transforming the graphical patterns into a CPN model, (4) automatically transforming the CPN model to the XML code executable by CPN Tools and validating the pattern semantics, (5) automatically generating the Esper EPL code and deploying it in a particular event-based system, and (6) providing a quantitative analysis of complex events through the CPN Tools executable model automatically generated by the tool. As future work, the authors plan to add additional features and functionalities to MEdit4CEP-CPN, such as further EPL operators and new transformation techniques.

Data and computational intelligence are outstanding research issues in the field of computer sciences which combined together represent one of the most emerging topics at present. We sincerely hope that you enjoy this special issue. We also have hopes that paper collection as a whole can pleasantly introduce readers to the composite and challenging arena of the application of computational intelligence to the scientific programming field, giving a fresh view of several state-of-the-art solutions from diverse perspectives. All accepted papers are within the scope of the journal and particularly the special issue, and all of them provide relevant and interesting research techniques, models, and work directly applied to the area of scientific programming. Before concluding, we want to express our sincere gratitude to all the authors who submitted their paper to this special issue and the many reviewers whose dedicated efforts made this special issue possible.

Conflicts of Interest

The editors declare that they have no potential conflicts of interest.

Acknowledgments

The editors wish to acknowledge the collaborative funding support from Ministerio de Economía e Innovación (Spain) (RTI2018-098652-B-I00 (MINECO/ERDF, EU) project), Ministry of Economy and Competitiveness (Spain) (CoSmart TIN2017-83964-R project), and Consejería de Economía e Infraestructuras/Junta de Extremadura (Spain)–European Regional Development Fund (ERDF) (GR18112 project and IB16055 project). The editors would also like to thank the reviewers for their generous time in providing detailed comments and suggestions that helped us to improve the quality of this special issue.

Álvaro Rubio-Largo
Juan Carlos Preciado
Luis Iribarne

References

[1] C. L. P. Chen and C.-Y. Zhang, "Data-intensive applications, challenges, techniques and technologies: a survey on big data," *Information Sciences*, vol. 275, pp. 314–347, 2014.

[2] S. Sun, C. Luo, and J. Chen, "A review of natural language processing techniques for opinion mining systems," *Information Fusion*, vol. 36, pp. 10–25, 2017.

[3] H. Harb, A. Makhoul, and C. Abou Jaoude, "A real-time massive data processing technique for densely distributed sensor networks," *IEEE Access*, vol. 6, pp. 56551–56561, 2018.

[4] L. Cao, "Data science: a comprehensive overview," *ACM Computing Surveys*, vol. 50, no. 3, pp. 1–42, 2017.

[5] K. Lepenioti, A. Bousdekis, D. Apostolou, and G. Mentzas, "Prescriptive analytics: literature review and research challenges," *International Journal of Information Management*, vol. 50, pp. 57–70, 2020.

[6] A. L'Heureux, K. Grolinger, H. F. Elyamany, and M. A. M. Capretz, "Machine learning with big data: challenges and approaches," *IEEE Access*, vol. 5, pp. 7776–7797, 2017.

[7] B. P. L. Lau, S. H. Marakkalage, Y. Zhou et al., "A survey of data fusion in smart city applications," *Information Fusion*, vol. 52, pp. 357–374, 2019.

[8] Y. Jin and B. Hammer, "Computational intelligence in big data [guest editorial]," *IEEE Computational Intelligence Magazine*, vol. 9, no. 3, pp. 12-13, 2014.

[9] Z. Lv, H. Song, P. Basanta-Val, A. Steed, and M. Jo, "Next-generation big data analytics: state of the art, challenges, and future research topics," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 4, pp. 1891–1899, 2017.

Hindawi
Scientific Programming
Volume 2019, Article ID 2604148, 17 pages
https://doi.org/10.1155/2019/2604148

Research Article

Facilitating the Quantitative Analysis of Complex Events through a Computational Intelligence Model-Driven Tool

Gregorio Díaz,[1] **Hermenegilda Macià**,[1] **Valentín Valero**,[1] **Juan Boubeta-Puig**,[2] **and Guadalupe Ortiz**[2]

[1]*School of Computer Science, University of Castilla-La Mancha, Campus Universitario s/n, 02071 Albacete, Spain*
[2]*Department of Computer Science and Engineering, University of Cádiz, Avda. de La Universidad de Cádiz 10, 11519 Puerto Real, Cádiz, Spain*

Correspondence should be addressed to Gregorio Díaz; gregorio.diaz@uclm.es

Received 15 February 2019; Revised 10 May 2019; Accepted 2 July 2019; Published 29 July 2019

Guest Editor: Alvaro Rubio-Largo

Complex event processing (CEP) is a computational intelligence technology capable of analyzing big data streams for event pattern recognition in real time. In particular, this technology is vastly useful for analyzing multicriteria conditions in a pattern, which will trigger alerts (complex events) upon their fulfillment. However, one of the main challenges to be faced by CEP is how to define the quantitative analysis to be performed in response to the produced complex events. In this paper, we propose the use of the MEdit4CEP-CPN model-driven tool as a solution for conducting such quantitative analysis of events of interest for an application domain, without requiring knowledge of any scientific programming language for implementing the pattern conditions. Precisely, MEdit4CEP-CPN facilitates domain experts to graphically model event patterns, transform them into a Prioritized Colored Petri Net (PCPN) model, modify its initial marking depending on the application scenario, and make the quantitative analysis through the simulation and monitor capabilities provided by CPN tools.

1. Introduction

Complex event processing (CEP) [1] is a computational intelligence technology used to analyze and correlate big data streams in order to detect situations of interest in real time. Events can be generated from other ones by matching the so-called *event patterns*, which are templates describing the conditions to be met to recognize such situations.

CEP fits in event-driven service-oriented architectures (SOA 2.0), in which the communications between applications and services are conducted by the complex events produced upon pattern detection. The event patterns to be detected for a particular application domain are implemented by using event-processing languages (EPLs) [2]. The MEdit4CEP approach [3] was proposed to help domain experts with this implementation. This framework provides an editor with graphical modeling capabilities for easily specifying the CEP domain, event patterns and action definitions. This approach generates Esper EPL code [4] from the graphical models. Moreover, MEdit4CEP was extended by using the Prioritized Colored Petri Net (PCPN) formalism [5], and the new version was called MEdit4CEP-CPN [6, 7]. In this framework, graphic event pattern models are automatically transferred into its graphical PCPN counterpart, which is transferred again into a compatible PCPN code interpretable by CPN Tools [8] to execute the event patterns, with the aim to perform the semantic validation of them. Thus, this approach offers the advantage of providing a graphical model to perform such a validation: users benefit from a better understanding of the internal semantics of the model through visual simulation of the model, where they can see its dynamic behavior and locate possible errors, for instance, in a certain condition, operation, time window, etc. Therefore, users not only observe whether there are differences between expected and actual outputs, but they can also inspect the internal dynamics of pattern execution when these differences occur.

Although CEP is so powerful for recognizing complex events in real-time big data streams, domain experts must face how to define the quantitative analysis to be performed in response to the produced complex events. These quantitative analysis studies are carried out to evaluate existing or planned scenarios, to compare alternative scenarios or to find an optimal scenario. In particular, we will focus on time-dependent scenarios since CEP systems consist of event streams flowing in time.

The main aim of this paper is, therefore, to demonstrate how MEdit4CEP-CPN can be used for conducting such quantitative analysis of events of interest for an application domain, without requiring knowledge of any scientific programming language for implementing the pattern conditions. Thereby, end users are provided with an all-in-one tool for graphically modeling event patterns, transforming them into a PCPN model, modifying its initial marking depending on the application scenario, and making the quantitative analysis through the monitor capabilities provided by CPN Tools. Obviously, the use of CPN Tools requires some knowledge from users in order to conduct the quantitative analysis, at least for modifying the initial marking of the produced CPN model and then executing the simulations to obtain the results. As indicated in our plans for future work, we intend to alleviate this problem by enriching our graphical model for event pattern design in order to be able to set the initial conditions (event flow) at design time and adding the option to automatically execute the produced CPN. The obtained output would then be transformed into the corresponding complex events in the output flow.

The quantitative analysis will be done by simulation and will involve statistical investigation of output data (complex events), exploration of large data sets, proper visualization of those output data, and the validation of simulation experiments. The outputs obtained from the simulations depend on the input data (simple events) that feed the system. These input data are set up stochastically, according to a specific scenario model. Thus, appropriate statistical techniques can be used for both designing and interpreting simulation experiments.

The structure of the paper is as follows. Section 2 depicts a general background describing the technologies and tools used in this work. Section 3 specifies the different steps followed in this work to perform the quantitative analysis. A case study about the sick building syndrome is then presented in Section 4, with a quantitative analysis using our methodology. Section 5 presents the related works, and a comparative study with our framework. Finally, Section 6 presents the conclusions and lines of future work.

2. Background

In this section, we introduce the main technologies used in this work, CEP and Colored Petri Nets (CPNs), and a brief description of the MEdit4CEP-CPN tool.

2.1. Complex Event Processing. CEP is a technology that captures, analyzes, and correlates large amounts of simple events with the ultimate goal of detecting relevant or situations of interest in a particular domain. Captured events are data that can flow through information systems, be provided by devices such as sensors, or come from social networks, among others. Such data are called simple events because they are characterized by being mainly raw data. The possibility of processing such simple events will allow us to infer information with a greater degree of semantic knowledge, thus obtaining the so-called complex events. For instance, for a stockbroker, the fact that the shares of a certain company fall 2% may be insignificant; however, that in a short period of time, both the shares fall 2% and also news about the low solvency of the company are published in a newspaper of economy; it could mean that it would be appropriate to sell the shares as soon as possible. In this case, the simple events would be that within a period of time t, the shares of the company x drop at least 2% and there is a negative news about the company in economic press. The complex event would be the recommendation of immediate sale. This way, in a given context, we will be able to detect the situations that are specifically relevant in that context or application domain. In order to do this, as Figure 1 shows, it will be necessary to previously define a series of event patterns specifying the conditions that simple input events must satisfy to detect such a situation. These patterns are defined and deployed in a CEP engine—software used to match these patterns on the incoming event flows, capable of analyzing the data and providing situations of interest detected in real time. The main advantage of CEP compared to other traditional event analysis software is the added ability to process large amounts of data and notify situations of interest detected in real time, allowing reduction of considerably the latency in decision making. This decision making capability relies on the architecture where the CEP engine is deployed, allowing the system either to perform certain actions or to enact certain measures.

As previously mentioned, CEP is a technology that can be very useful in several application domains such as financial systems, health care, energy optimization, online sales and marketing, business intelligence, security, and transportation, among many others, since CEP objective is to offer a general paradigm to be applied to a great variety of systems [1, 9–14]. However, a deep knowledge of the application domain is required to be able to define the patterns that may be relevant to that domain depending on the simple events that can be obtained in the system. Companies have normally domain experts and computer science experts, but these skills usually do not fall on the same person and the definition of patterns in the language provided by CEP engines is not trivial. For this reason, in the past, we proposed MEdit4CEP [3].

MEdit4CEP was defined and implemented for the purpose of providing a tool for CEP pattern definitions appropriate for domain experts with no particular programming skills. Thus, MEdit4CEP is a model-driven solution for real-time decision making in SOA 2.0 that provides a graphical modeling editor for CEP domain definition and a graphical modeling editor for event pattern

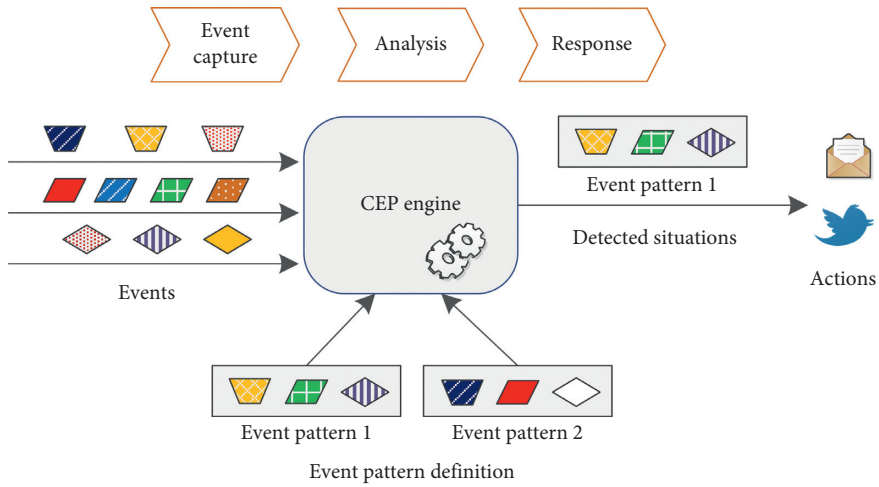


Figure 1: Complex event-processing stages.

definition, as well as automatic code generation and deployment from the patterns modeled by the domain expert.

2.2. Petri Nets and Quantitative Analysis. A Petri Net (PN) is defined as a bipartite-directed graph which has two types of nodes, places (depicted as circles) and transitions (depicted as rectangles), connected using arcs between either places and transitions (pt-arcs) or transitions and places (tp-arcs) [15]. Places of a Petri Net are used to represent system states and conditions, and a transition represents an action or an event producing a change in the system state.

Definition 1 (Petri Net). A Petri Net is a triple (P, T, F), where P is the set of places, T is the set of transitions, $X = P \cup T$ is the set of nodes, and $F \subseteq (P \times T) \cup (T \times P)$ is the set of arcs. For any node $x \in X$ (place or transition), we define the preconditions and postconditions of x, denoted by ${}^\bullet x$ and $x^\bullet$, respectively, as follows: ${}^\bullet x = \{y \in X \mid (y, x) \in F\}$, $x^\bullet = \{y \in X \mid (x, y) \in F\}$.

The dynamic evolution of a PN is captured by the so-called *markings*. A marking of a PN is a function $M: P \longrightarrow \mathrm{IN}$, which assigns a natural number to each place. This number is usually indicated as a set of dots inside the place or by a number beside the place, and it is called the number of *tokens* on the place. This number can be used for instance to indicate the number of events that must be processed or the number of processes in a queue.

A natural number (*arc weight*) is used to label a pt-arc. This number indicates how many tokens are required to fire (*execute*) an outgoing transition, by default, one. Tp-arcs also have a weight associated, which specifies how many tokens will be produced at the outgoing place when the transition is executed. A transition t can then be fired (*enabling condition*) when all its precondition places have at least as many tokens as the weight of the arc that connects them to t. The firing of a transition t removes from its precondition places a number of tokens equal to the weight of the pt-arc (p, t) connecting them and inserts new tokens on its postcondition places, according to the weights of those tp-arcs.

Petri nets are untimed and tokens do not carry any information. CPN [16] is a PN extension which incorporates data and time, allowing to model complex data structures attached to tokens. Thus, places have an associated *color set* (a data type), indicating the set of permitted token colors at a given place.

CPN Tools [8] is a widely used tool that allows us to create, edit, simulate, and analyze CPNs. The notation described below is the one used in this tool. In CPNs, we can have places with no attached information (color set *UNIT*), as in the plain model. But we can have other color sets, such as the set of integer numbers *INT*, a Cartesian product of two or more color sets as $INT2 = INT \times INT$, $INT3 = INT \times INT \times INT$, and a string ($STRING$). Each token has then an attached data value *(color)*, which belongs to the corresponding place color set.

In CPN Tools, the current number of tokens on every place is drawn in green beside the place circle, and the specific colors of these tokens are indicated using the notation $m'v$, meaning that we have m instances of color v. When we have several tokens on a place with different values, we use the symbol "++" to represent the union of them.

Arcs can have inscriptions (*arc expressions*), constructed using variables, constants, operators, and functions, whose evaluation matches the *color set* of the attached place. For a transition t with variables $y1, y2, \ldots$ in its input arc expressions, a *binding* of t is an assignment of specific values to each of these variables. A transition t is then *binding enabled* if there is a *binding* such that the evaluation of each input arc expression of t matches the corresponding tokens (with the same values) in the corresponding input place.

We can have guards associated to transitions, which can be used to restrict their firing. Guards are predicates constructed by using the variables, constants, operators, and functions of the model. For a guarded transition to be *fireable*, the evaluation of the guard must be true with the selected binding. A priority can also be associated to a transition. When two or more transitions can be fired (executed) at a given time, the transition with the highest

level of priority is fired first. A CPN with priorities is called a prioritized CPN (PCPN). Specifically, we use the following priorities: P_HIGH, P_NORMAL, P_LOW, P_LOW_1, P_LOW_2, ... P_LOW_n (for a certain $n \in \mathrm{IN}$) and P_MIN, following this decreasing order of priority.

CPN Tools allows us to split the model into pages, which is a useful feature to deal with large models. In this case, both substitution transitions and fusion places can be used to conform the whole model. Substitution transitions allow us to create hierarchical models, in which some transitions represent the actions enclosed in other CPN pages, while fusion places are places that are used in different pages, i.e., the places identified by a same fusion label are functionally the same place.

Example 1. Let us consider the PCPN depicted in Figure 2. Places *InputEv* and *OutEv* have *INT2* as color set and *ProcSqn* and *SqOut* have *INT* as color set. The initial marking of both places *ProcSqn* and *SqOut* is $1'1$ (one integer token with value 1). Place *InputEv* represents a flow of input events of a system that must process these events producing as output a flow with those events whose value is greater than 2 (place *OutEv*). Each event is represented by a Cartesian product of 2 integers (colorset *INT2*), in which the first component stands for the event number and the second component for the event value (integer). The initial marking of place *InputEv* is $M = 1'(1,1) + +1'(2,4) + +1'(3,0) + +1'(4,9)$, as shown in the figure. Transitions are labeled with their associated guard and priority information (P_NORMAL if empty), and arcs are labeled with their corresponding expressions. All the variables used in the expressions (n, m, v, k) are integers.

In this PCPN, place *SqOut* allows us to number sequentially the events produced on *OutEv*. Place *ProcSqn* acts as a sequence counter so as to process the event tokens on *InputEv* in order. Transition *selcond* must be fired when we have one token (n, v) on *InputEv* with a sequence number n equal to that indicated on *ProcSqn* that fulfills the condition $v > 2$. Transition *selcond* updates the sequence number on *ProcSqn*, by increasing it by one unit. Otherwise, when transition *selcond* cannot be fired, transition *incr_sq* is fired in order to increase the sequence number on *ProcSqn*, but it will stop firing when the sequence number on *ProcSqn* is greater than the maximum sequence number on *InputEv*.

The final marking obtained on the place *OutEv* is therefore $M' = 1'(1,4) + +1'(2,9)$ for the initial marking indicated in the figure. The final marking on *ProcSqn* is $1'5$ and on *SqOut* is $1'3$, and place *InputEv* keeps its initial marking.

Quantitative analysis in CPNs allows us to obtain relevant performance indexes of the system modeled. For instance, this analysis is used to obtain average response times, throughput, queue lengths, etc. In our case, the quantitative analysis can be used both to validate the event patterns defined and also to obtain predictive information by feeding the system with different event scenarios. Quantitative analysis using CPNs is usually based on simulations in order to obtain the measures of interest for the modeled scenario. This simulation-based quantitative analysis is performed through a number of lengthy simulations of a CPN model, during which data are collected from the occurring binding elements, firing of transitions, and markings reached so as to obtain estimates of measures of interest; in our case, the expected outputs of the system. This information is gathered by repeating the same experiment (simulation) a number of times, using the replication capabilities of CPN Tools and then using the monitoring capabilities of CPN Tools to extract the relevant data from the simulations. Specifically, we use place content break point and data-collector monitors, which allow us to determine whether a place becomes marked and extracts numerical data during simulations, respectively. For instance, these monitors can be used to count the number of times a specific transition has been fired across a simulation, to extract the marking of some specific places of the CPN model or to obtain the first instant at which a specific transition was fired.

Thus, the quantitative analysis is basically based on the monitor and replication capabilities of CPN Tools, which provides us with a report and log files which can be analyzed by using other well-known statistical computing tools, such as *R*, Matlab, and SPSS.

As an illustration, let us consider the CPN depicted in Figure 2. It might be of interest to know how many times transition *selcond* is fired, i.e., the amount of tokens gathered at place *OutEv*. Figure 3(a) shows the binder tools palette of CPN Tools for monitoring purposes, and Figure 3(b) shows the monitor that has been defined to count the number of firings of transition *selcond*. In this case, as it can be seen from the example, the result obtained by applying the monitor was 2.

Experiments can then be produced for different scenarios by modifying the initial marking, which can also be randomly generated so as to produce synthetic scenarios. For instance, we could define a function *M_init* to produce a random initial marking with n tokens for place *InputEv* in Figure 2 using a discrete uniform distribution, as follows:

$$\begin{aligned}\mathrm{funM\_init}(n) = {} & \mathrm{if}\,(n = 0)\,\mathrm{then\ nil\ else}\ 1'(n,\ \mathrm{discrete}(1,6)) \\ & ++\mathrm{M\_init}(n-1).\end{aligned} \tag{1}$$

To reproduce the experiments, we can use the replica capabilities provided by CPN Tools. The following expression simulates m times this example:

$$\mathrm{CPN'Replications.nreplications\ m}. \tag{2}$$

The outputs obtained for these experiments using these stochastic initial markings can then be analyzed with the statistical computing tools mentioned above.

2.3. MEdit4CEP-CPN. As previously explained, this tool was introduced in [6] as an extension of MEdit4CEP [3] to deal with the semantic validation of the modeled patterns.

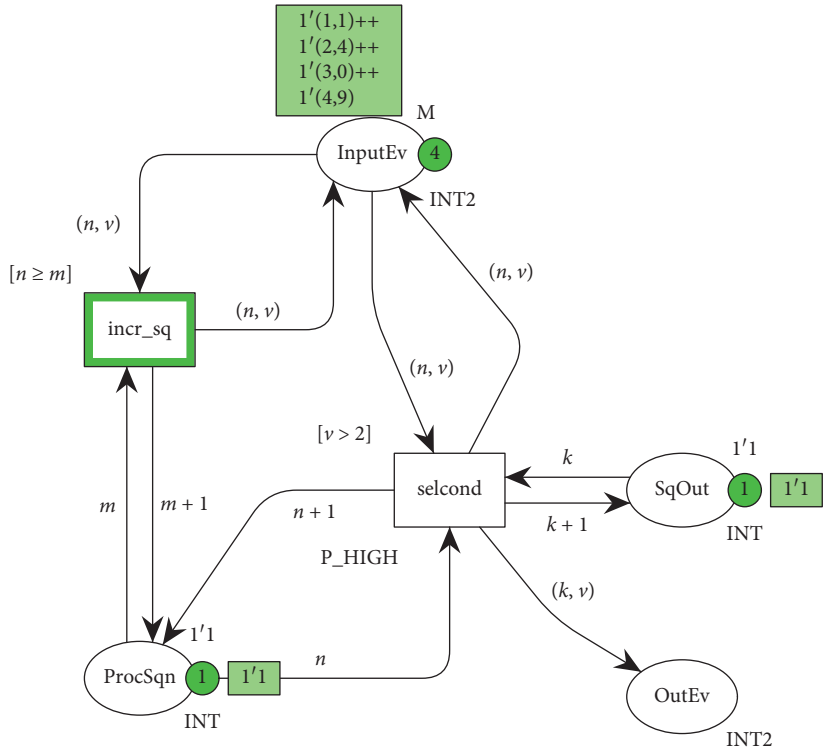
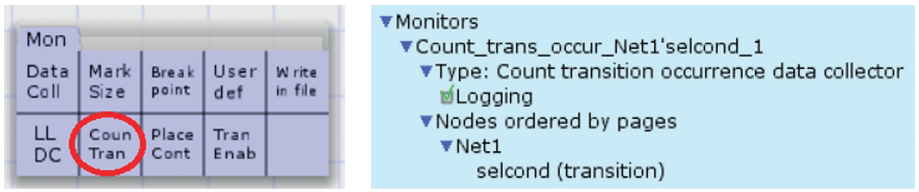


Figure 2: Graphical view of a marked PCPN.



(a) (b)

Figure 3: Monitor use in CPN Tools. (a) Monitor example. (b) Monitor example.

In particular, MEdit4CEP-CPN mainly consists of a domain-specific modeling language (DSL) and a graphical modeling editor for automatically transforming event pattern models into PCPN graphical models. Then, these models are validated and transformed into codes executable by Petri nets software.

This DSL was implemented using the Epsilon languages [17] for model-to-model transformation, model validation, and template-based code generation. Additionally, Epsilon EuGENia [18], a front-end for the graphical modeling framework, was used for implementing the editor. More details about the implementation can be found in [6].

Figure 4 illustrates the 7 phases, explained below, a user can follow to accomplish not only the semantic validation of the modeled patterns but also to be able to perform a quantitative analysis of the complex event properties in the studied scenarios.

2.3.1. Event Pattern Model Definition. In phase 1, the tool user is expected to graphically define the event patterns to be detected in a particular application domain.

2.3.2. Event Pattern Model Syntactic Validation. Once an event pattern has been modeled (phase 1), thanks to the use of the presented editor, the user can automatically validate the pattern syntax (phase 2). The editor will check whether the model conforms to the ModeL4CEP metamodel. Afterwards, the errors to be fixed before continuing will be shown. As of this phase, we can accomplish a semantic validation through PCPNs (phases 3, 4, 5, and 6); otherwise, phase 7 can be performed with the aim of automatically transforming the model into EPL code.

2.3.3. Event Pattern Model Transformation to PCPN Model. In phase 3, the event pattern models are automatically transformed into a PCPN model. In order to provide such a functionality, the editor has been provided with a metamodel for PCPN and a set of model-to-model transformation rules that we have defined and implemented for this purpose. Thus, a PCPN conforming to the named metamodel is generated.

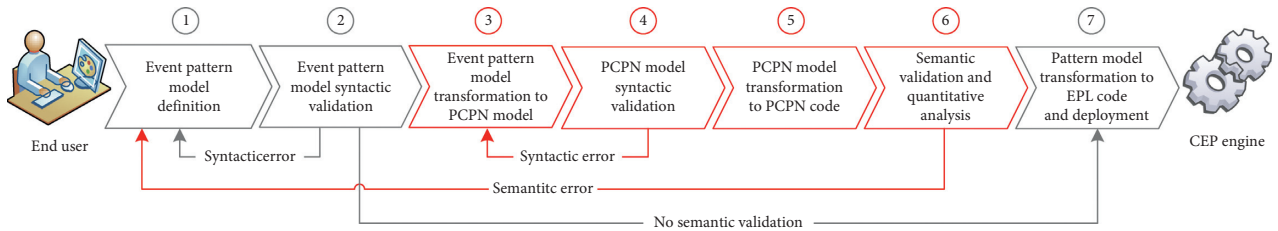


Figure 4: MEdit4CEP-CPN: a model-driven approach for CEP modeling by PCPN.

2.3.4. PCPN Model Syntactic Validation. Once the PCPN model has been automatically generated, in phase 4, domain experts may modify the PCPN according to their needs. For instance, they might be interested in editing the initial marking to check other particular scenarios of their interest. Then, after the PCPN edition, (1) it is checked whether the new model conforms to the PCPN metamodel and (2) whether the validation rules are satisfied through a syntactic validation. The errors that should be fixed before continuing with the following phase would be shown at this stage.

2.3.5. PCPN Model Transformation to PCPN Code. In phase 5, the PCPN model is automatically transformed into executable PCPN code (PCPN code refers to a proprietary PCPN file format that can be executed by a specific software); a set of model-to-text transformation rules have been defined and implemented for this purpose.

2.3.6. Semantic Validation and Quantitative Analysis. The expert in charge of simulating and analyzing the PCPN will then feed the net with an arbitrary number of initial markings (stream of events) in phase 6. This way, it will be possible to check if the event pattern is semantically correct, as well as a quantitative analysis will be carried out (see Section 3). In the case a semantic error is detected, we should return to phase 1.

2.3.7. Pattern Model Transformation to EPL Code and Deployment. Finally, in phase 7, the event pattern model is automatically transformed into EPL code and deployed in the CEP engine in question. In this work, we are generating code for the Esper CEP engine, but new transformation rules for other CEP engines of interest may be easily created and integrated in the proposed editor.

Therefore, we can conclude that, we have a top-down approach in which users can graphically define what they want to model (event patterns) and the proposed system automatically provides the implementation code. In this way, according to the capabilities associated to phases 5 and 6, MEdit4CEP-CPN allows us to infer additional meaningful information and to obtain predictive results about the analyzed pattern by feeding the system with different initial scenarios (markings).

3. Quantitative Analysis of Complex Events

CEP is a new class of event-processing solution which integrates into standard middleware architectures and enables event processing to be embedded in any standard enterprise application. This new service technology brings the power of event-driven insight into any industry and any end user. But sometimes, it is not easy to be used by domain experts. In this sense, MEdit4CEP-CPN was created to reach the next objectives in the CEP scope:

(i) Creation of user-friendly models for facilitating domain experts the task of defining event patterns

(ii) Syntactic and semantic model validation through model-driven techniques and CPN Tools, respectively, without requiring deep knowledge on CEP or PCPN formalism

(iii) Model-to-model and model-to-text transformations for automatically transforming the event pattern models into both PCPN models and EPL code and also the PCPN models into code executable by CPN Tools

(iv) Plug-in based solution, what will allow us to easily extend it with additional capabilities such as glass box and black box testing techniques for testing models, or predictive analytics for predicting future scenarios by analyzing the historical data

In this paper, we focus on phase 6 (semantic validation and quantitative analysis), briefly described in Section 2.3, which receives as an input the automatically generated PCPN code executable by CPN Tools. In this paper, more specifically, we define and carry out the particular phases that must be followed to conduct the quantitative analysis of the system under study (see Figure 5).

3.1. 6(a) Scenario Configuration. The initial marking (M_0) of the generated PCPN is initialized with an ordered flow of simple events ($F = e_1, e_2, e_3, \ldots$), representing a specific scenario. This event flow can be introduced manually or generated automatically by using deterministic or probability distribution functions provided by CPN Tools. Note that the automatic data generation is very convenient for analysis purpose.

3.2. 6(b) Deterministic Quantitative Analysis. The PCPN is then executed using CPN Tools in order to obtain the

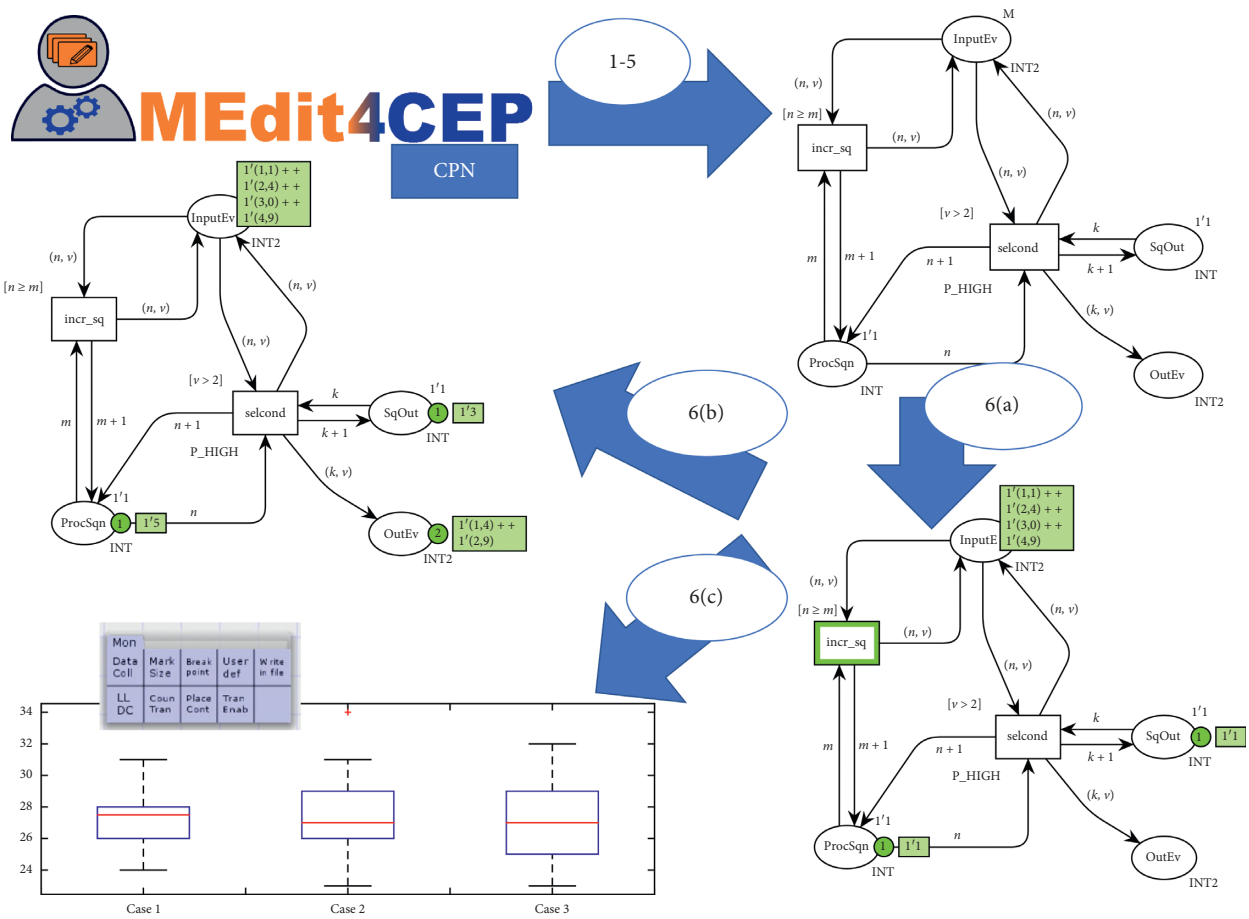


Figure 5: Phases to conduct the quantitative analysis.

corresponding output (detected complex events). Thus, this phase allows us to conduct the semantic validation as well as the quantitative analysis with a deterministic input event flow.

3.3. 6(c) Stochastic Quantitative Analysis. Alternatively to phase 6(b), the initial marking M_0 is obtained using stochastic input event flows, according to different scenarios, and the PCPN is then executed using simulations and the monitor capabilities provided by CPN Tools.

Therefore, the analysis of different scenarios is performed in phase 6, in which we feed the PCPN with initial markings that simulate the scenarios to be studied. In particular, we can produce the initial markings by using either deterministic functions that simulate specific situations or using probability distributions that produce stochastic values reproducing scenarios of interest. For instance, a normal distribution could be used to produce increments in a temperature measure, an exponential distribution to represent the users arriving at a hospital emergency room, and so on. In this case, when using stochastic initial markings, CPN Tools allows us to replicate the execution of a CPN by automatically producing the initial markings, and the monitor capabilities can then be used to collect the relevant data from the simulations.

4. Case Study: Sick Building Syndrome

As an illustration of the methodology described in the previous section, we consider a scenario of events related to the sick building syndrome (SBS), which is considered by the World Health Organization (WHO) [19] as a group of symptoms that people suffer in a building for no apparent reason. Some of the SBS symptoms are nose, throat, and eye irritation; itching, dry and red skin; dry mucous membranes sensation; mental fatigue; and headaches, and dizziness, and nausea. These symptoms tend to increase in severity as people spend more time in the building but get reduced over time or even disappear when people are away from the building. Apart from this health problem, people's work performance becomes obviously affected, with a corresponding loss of productivity. The SBS is widespread and may occur in hospitals, offices, apartment houses, nurseries, schools, and so on. Although the cause of SBS is unclear, some main factors related to indoor air quality (IAQ) are chemicals emissions from different sources, particles, radon, pets and pests, microbes, temperature, humidity, and ventilation.

In this sense, the WHO provides the guidelines [20] for the protection of public health from risks for some selected pollutants commonly present in indoor air, including the carbon monoxide, which is the pollutant that we consider in this work.

These are the recommendations related to indoor exposures of *CO*:

(i) 100 mg/m^3 for 15 minutes (assuming light exercise and that such exposure levels do not occur more often than one per day)

(ii) 35 mg/m^3 for 1 hour (assuming light exercise and that such exposure levels do not occur more often than one per day) (30 mg/m^3 is also recommended in some European Agencies such as https://www.anses.fr/fr/system/files/AIR2004etVG003Ra.pdf)

(iii) 10 mg/m^3 for 8 hours (arithmetic mean concentration and light-to-moderate exercise)

(iv) 7 mg/m^3 for 24 hours (arithmetic mean concentration, assuming that the exposure occurs when people are awake and alert, but not exercising)

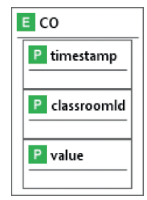
Let us consider, for instance, a school as the specific indoor scenario, where children stay in a classroom for 5 h/day. We then focus on the second recommendation: a nonhealthy indoor air quality corresponds to an average level greater than or equal to 35 mg/m^3 of *CO* for 1 hour.

Next, we follow the phases described in Section 2.3 together with the new phases proposed in this paper (see Section 3) in order to make the quantitative analysis of the complex events detected for this scenario by using MEdit4CEP-CPN and CPN Tools.

4.1. Phases 1–5: Event Pattern Model Definition, Syntactic Validation, and Transformation to PCPN Model, as well as PCPN Model Syntactic Validation and Transformation to PCPN Code. The domain considered for this hypothetical scenario consists of *CO* measurements gathered every 5 minutes in a specific classroom. Thus, a simple event consists of a measure for the *CO* pollutant, the classroom identifier where the measure was gathered and the timestamp of the measure. We consider the event time stamps as integers (in minutes), classrooms identifiers as strings, and *CO* values as real numbers. Using MEdit4CEP-CPN, we can easily define this domain (*CO* event type). Figure 6(a) depicts the domain modeled and syntactically validated with MEdit4CEP-CPN, and Figure 6(b) shows its automatic translation to EPL code.

Patterns can be easily modeled and syntactically validated using the tool. Figure 7(a) shows the *CO_Avg*-modeled pattern, which computes the *CO* pollutant average during the last hour, while Figure 7(b) shows its automatic transformation into EPL code. In the same way, Figure 8(a) shows the *CO_Unhealthy* modeled pattern, to detect and recognize situations in which the average computed with *CO_Avg* is greater than or equal to 35 mg/m^3, and Figure 8(b) shows its transformation into EPL code.

Using our MEdit4CPN-CPN tool again, the *CO_Avg* and *CO_Unhealthy* event patterns are automatically transformed into their corresponding PCPN models consisting of four different pages (see [6] for a complete description of these transformations): two pages for the pattern transformation to obtain the *CO* pollutant average and the condition to establish whether the threshold value of 35 mg/m^3 has been reached or not (see Figures 9 and 10, respectively) and two pages for simple and complex events (see Figures 11 and 12, respectively). The corresponding CPN Tools declarations are also shown in Listing 1.1. For simplicity, we consider in the PCPN model that one unit time corresponds to 5 minutes, therefore tp_CO_Avg = 12 in Listing 1.1.



(a)

```
create schema CO(timestamp integer,
  classroomId string, value double);
```

(b)

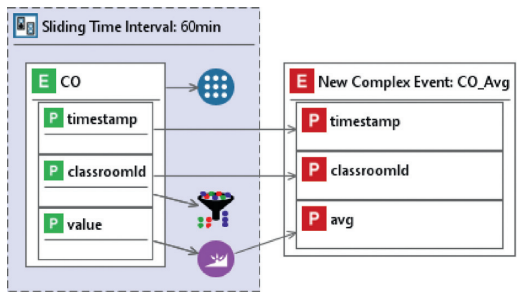
Figure 6: Modeled domain. (a) Domain in MEdit4CEPCPN. (b) Domain in EPL.

4.2. Phase 6(a): Scenario Configuration. Similarly to Example 1, the initial marking in place *CO_in* contains the tokens that represent the input events (see Figure 13). In this case, these events represent the *CO* values that have been measured. In the scenario considered, for the initial marking of the *CO_in* place, we start with a *CO* value of 3.0 mg/m^3. This value is then increased by a value of 0.5 mg/m^3 every 5 minutes.

4.3. Phase 6(b): Deterministic Quantitative Analysis. After executing the PCPN model by running a simulation, a value of 30.25 mg/m^3 is reached after 60 time units (300 minutes) in the *CO_Avg* place (see Figure 14), so the *CO_Unhealthy* place is empty, because no complex event has been produced corresponding to this situation. As a consequence, in this scenario and with the values produced by this specific simulation, the threshold value of 35 mg/m^3 was not reached as the average level of *CO* computed in the considered period of time.

4.4. Phase 6(c): Stochastic Quantitative Analysis. This phase consists in simulating different scenarios by modifying the initial marking with stochastic input event flows, thus obtaining the quantitative results for those scenarios. CPN Tools provides a simulator engine, which allows us to automatically replicate simulations of a scenario using its monitor capabilities. This is an important advantage of using CPN Tools: we can obtain relevant performance measures through simulation experiments, using the monitor features of CPN Tools. As previously mentioned, monitors are used to observe, inspect, or control simulations. In particular, we use data-collector monitors, which are used to extract numerical data from a PCPN. The numerical data obtained are then used to compute the statistic information.

To accomplish this objective, we use synthetic data that represent different scenarios. Thus, a new CPN Tools page was included in the PCPN model to produce the initial markings for different scenarios (Phase 6(a)), starting from a certain value of *CO* pollutant and increasing it by using a probability

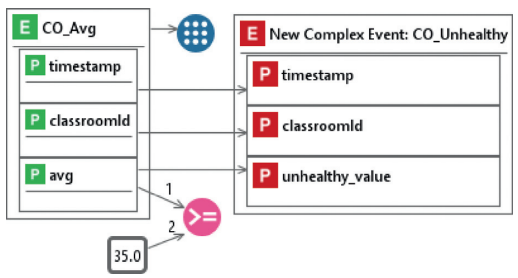


(a)

```
@Name('CO_Avg')
insert into CO_Avg
select a1.timestamp as timestamp,
  a1.classroomId as classroomId,
  avg(a1.value) as avg from
pattern [(every a1 = CO)].
  win:time(60 minutes)
group by a1.classroomId
```

(b)

Figure 7: CO_Avg event pattern. (a) CO Avg in MEdit4CEP-CPN. (b) CO Avg in EPL.



(a)

```
@Name('CO_Unhealthy')
insert into CO_Unhealthy
select a1.timestamp as timestamp,
  a1.classroomId as classroomId,
  a1.avg as unhealthy_value from
pattern [(every a1 = CO_Avg(a1.avg
  >= 35.0))]
```

(b)

Figure 8: CO_Unhealthy event pattern. (a) CO Unhealthy in MEdit4CEP-CPN. (b) CO Unhealthy in EPL.

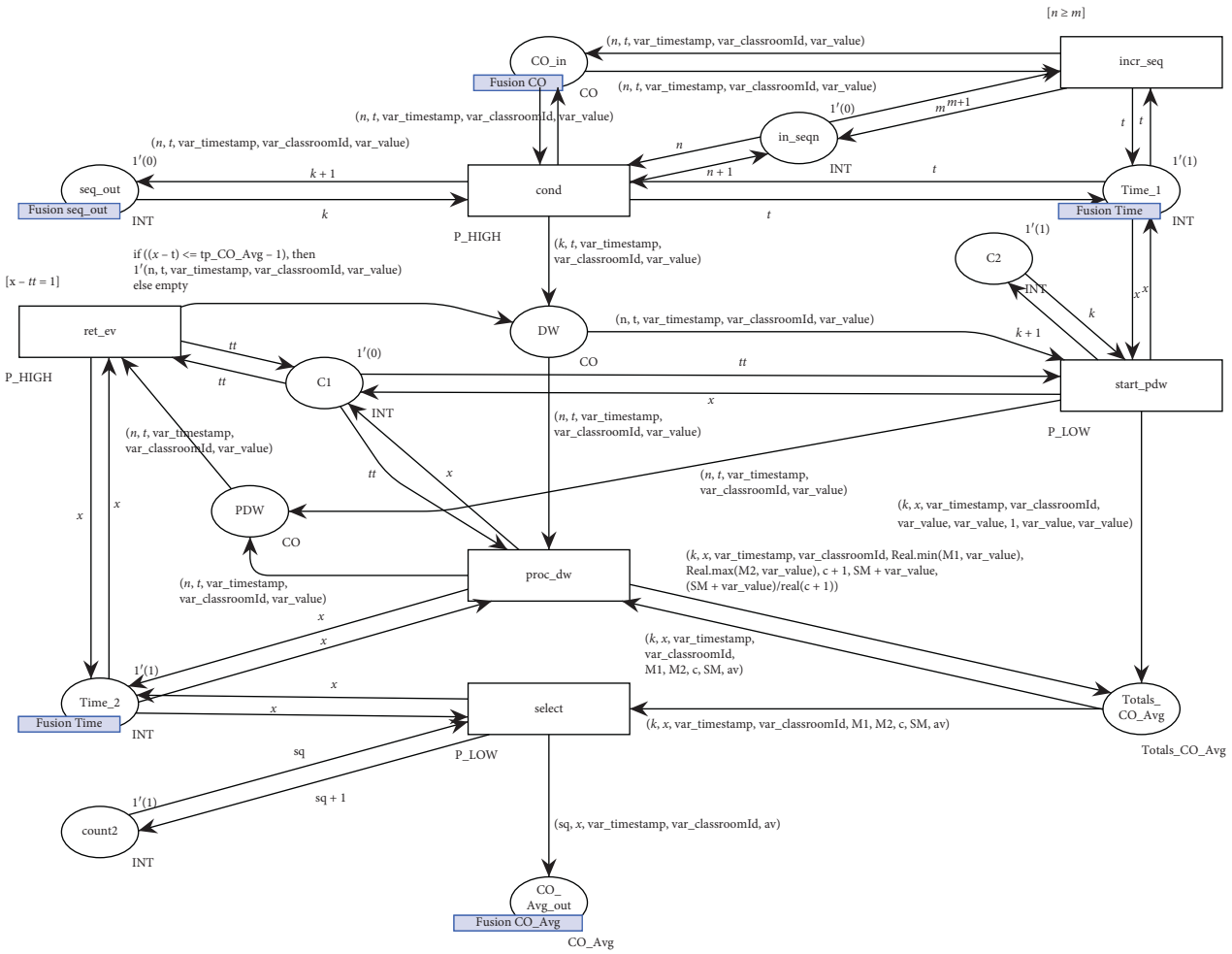


Figure 9: CO_Avg page.

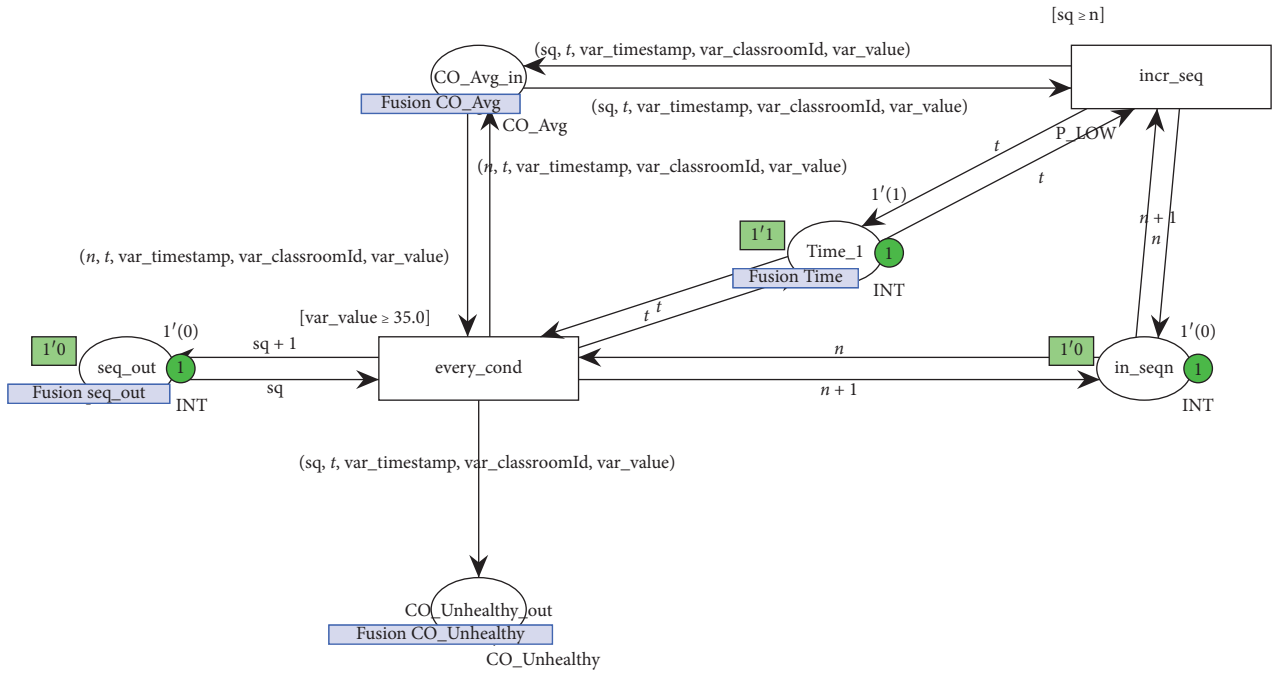


Figure 10: CO_Unhealthy page.

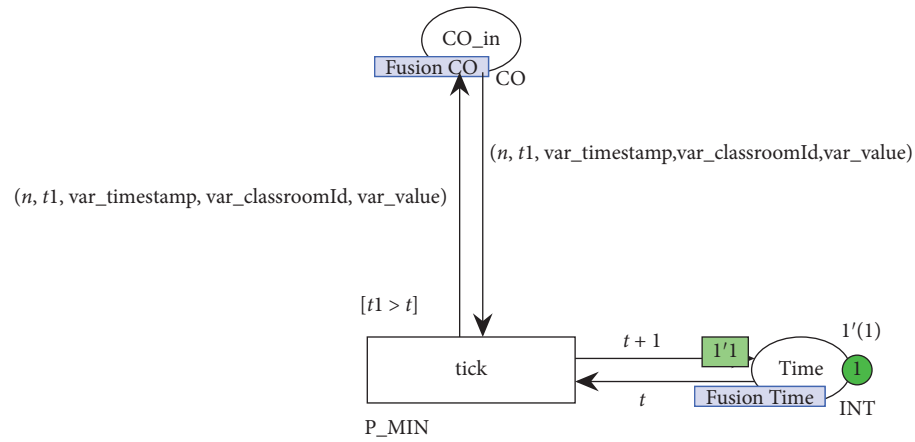


Figure 11: Events page.

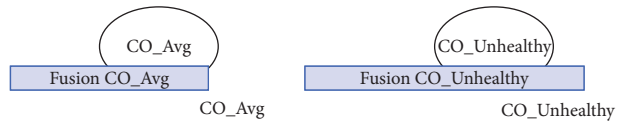


Figure 12: Complex events page.

distribution (see Figure 15). In this figure, transition *tinitial* fires 61 times in order to produce 61 events on place *CO_in* (we produce 61 events to allow the sliding time window to be processed for the first 60 events because we need the clock to reach the value 61). With each firing, the *CO* value (represented with the variable xx) on place *initial* is updated, by increasing it with a value obtained from a uniform distribution with arguments $f1$ and $f2$ (xx + uniform ($f1$, $f2$)). By changing the initial *CO* value, the parameters and/or the probability distribution function, we can easily generate different scenarios.

We now apply the monitor features of CPN Tools, which allow us to observe, inspect, control, or modify a simulation of a CPN. We consider two situations of interest in this work. The first checks whether the scenario reaches an unhealthy situation and the second the time of the first occurrence of this unhealthy situation. For this purpose, two monitors are specified, respectively. The first monitor (*reach_place_unhealthy*) is a place content break point monitor, and it will indicate us if there is at least one token in place CO_Unhealthy of Figure 14, and the second one (*time_first_unhealthy*) is a data-collector monitor, which

```
(1) (* Standard priorities *)
(2)    val P_MAX = 10;
(3)    val P_HIGH = 100;
(4)    val P_NORMAL = 1000;
(5)    val P_LOW = 10000;
(6)    val P_MIN = 20000;
(7) (* Standard declarations *)
(8)    colset INT = int;
(9)    colset STRING = string;
(10)   colset REAL = real;
(11)   colset BOOL = bool;
(12)   colset UNIT = unit;
(13)   colset TIME = time;
(14)   colset INTINF = intinf;
(15) (* Declarations for domain: SBS_CO *)
(16)   colset CO = product INT * INT * INT * STRING * REAL;
(17)   (* vars_for_event: CO *)
(18)      var var_timestamp: INT;
(19)      var var_classroomId: STRING;
(20)      var var_value: REAL;
(21)   var n, t, t1: INT;
(22) (* Declarations for complex events domain: SBS_CO *)
(23)   colset CO_Avg = product INT * INT * INT * STRING * REAL;
(24)   (* vars_for_complexevent: CO_Avg *)
(25)      var var_avg: REAL;
(26)   colset CO_Unhealthy = product INT * INT * INT * STRING * REAL;
(27) (* vars_for_complexevent: CO_Unhealthy *)
(28)   var var_unhealthy_value: REAL;
(29) (* Total color set for: CO_Avg *)
(30)   colset Totals_CO_Avg = product INT * INT * STRING * REAL * REAL * INT * REAL * REAL;
(31) (* Pattern auxiliary variables *)
(32)   var m, sq, k, tt, x, m1, m2, c, sm: INT;
(33)   var M1, M2, SM, av: REAL;
(34) (* Sliding time interval declarations: *)
(35)   val tP_CO_Avg = 12;
(36) (* Pattern auxiliary variables *)
(37) (* Declarations initial marking *)
(38)   colset INT3 = product INT * INT * INT;
(39)   var xx:REAL;
(40)   val co0 = 3.0;
(41)   val f1 = 0.5;
(42)   val f2 = 0.5;
```

Listing 1: CPN Tools declaration.

indicates us the time at which the place was reached for the first time. Both monitors are shown in Figure 16. In the first monitor, we only indicate the name of the place that will stop the execution if it becomes marked (*CO_unhealthy*), while in the second monitor, we need to write a predicate indicating the stop condition and an *Observer* function to obtain the time at which the execution stopped.

Once, the initial configuration is established, we can use the CPN Tools simulator engine to automatically replicate the experiment n times so as to obtain performance results. For instance, we can replicate it 100 times by using the following code:

$$\text{CPN}'\text{Replications.nreplications 100.} \quad (3)$$

Next, we introduce a specific case study, shown in Table 1, with 9 different scenarios. The objective of this study is to examine the probability of reaching an unhealthy condition before the 300 minutes. In these scenarios, the initial value for *CO* in all the simulations is (3.0 mg/m^3) and 100 replications have been used to obtain the results. As an illustration, in the third row corresponding to the third scenario, each time the *CO* value is updated in the initial place (see Figure 15), its value is increased by an arbitrary increment between 0.5 and 0.65. Therefore, the parameters for the uniform distribution function uniform ($f1$, $f2$) are established to $f1 = 0.5$ and $f2 = 0.65$. In this experiment, only 4% of the simulations reached an unhealthy situation. Notice that when the increment arguments are $f1 = 0.5$ and $f2 = 0.72$ (last row in the table), we always obtain an unhealthy situation (100%).

Other distribution functions can be considered as well. For instance, we can consider a normal distribution normal ($f1$, $f2$)

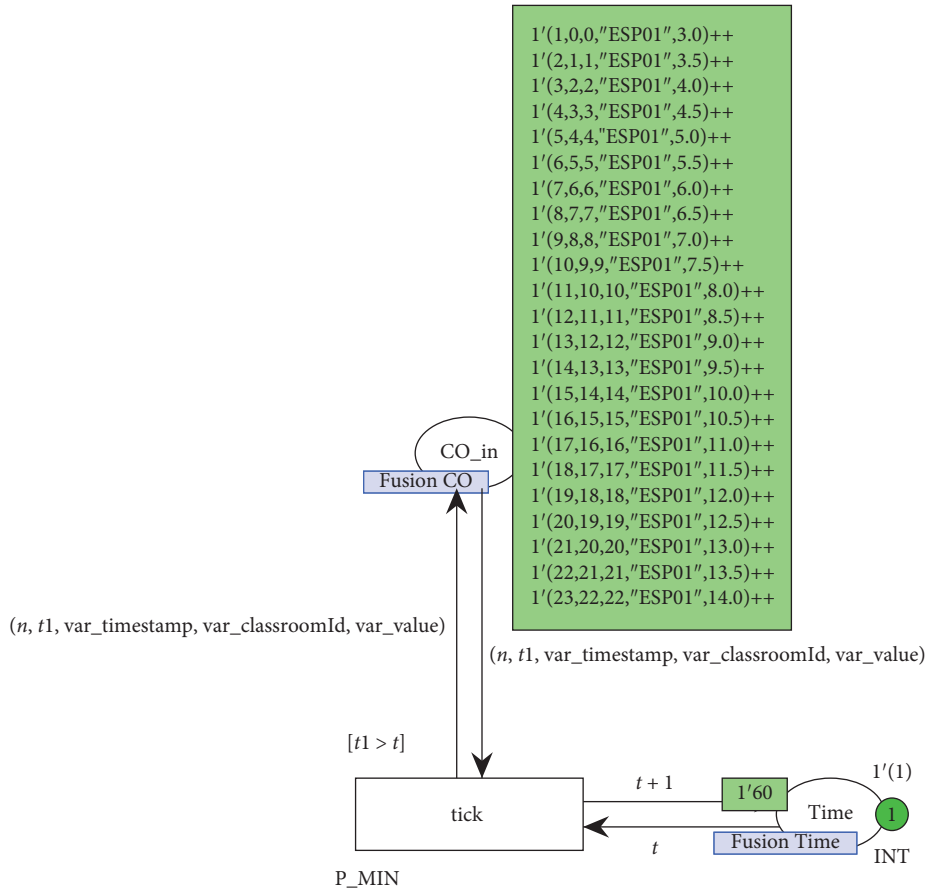


Figure 13: Initial marking in the event page.

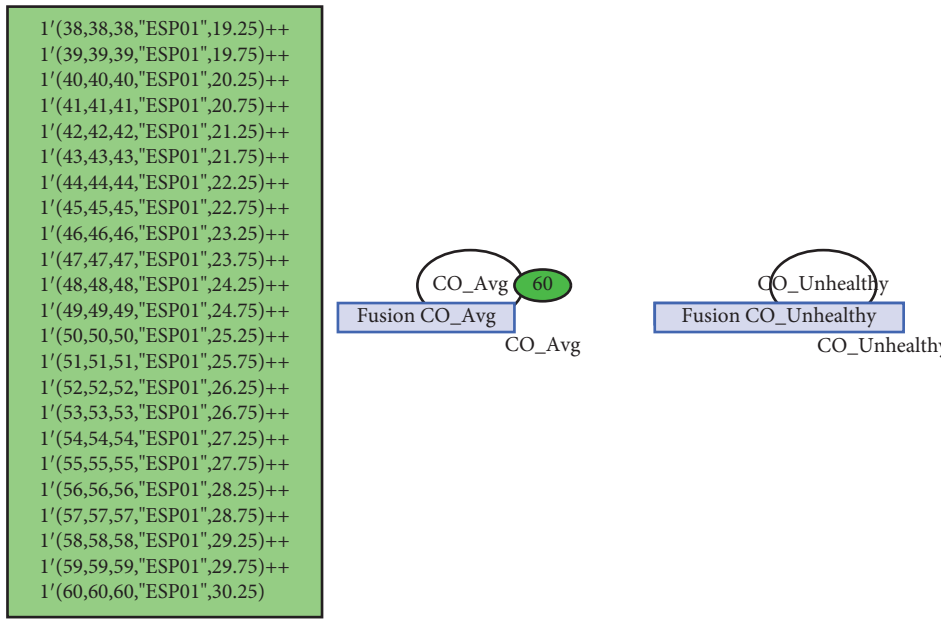


Figure 14: Final marking page.

to produce the events in the initial marking, as well as considering the initial value of CO of 3.0 mg/m^3. In particular, in the scenarios obtained for the normal distribution with parameters $f1 = 1.5$ and $f2 = 0.25, 0.5$, and 0.75, an unhealthy situation is always reached before 300 minutes. However, the time to reach this situation varies depending on the value of

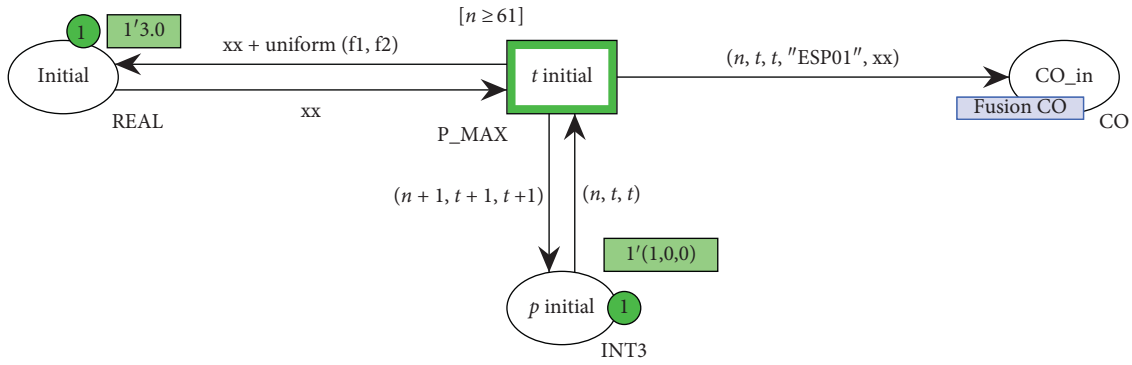


Figure 15: Initial marking page.

```
Monitors
  reach_place_unhealthy
    Type: Place content break point
      If is empty
    Nodes ordered by pages
      CO_Unhealthy
        CO_Unhealthy_out (place)
  time_first_unhealthy
    Type: Data collection
      Timed
      Logging
    Nodes ordered by pages
      CO_Unhealthy
        every_cond (transition)
    Predicate
      fun pred (bindelem) =
      let
        fun predBindElem (CO_Unhealthy'every_cond (1,
                          {n,sq,t,var_classroomId,
                          var_timestamp,var_value})) = true
          | predBindElem _ = false
      in
        predBindElem bindelem
      end
    Observer
      fun obs (bindelem) =
      let
        fun obsBindElem (CO_Unhealthy'every_cond (1,
                         {n,sq,t,var_classroomId,
                         var_timestamp,var_value})) = t
          | obsBindElem _ = ~1
      in
        obsBindElem bindelem
      end
    Init function
      fun init () =
        NONE
    Stop
      fun stop () =
        NONE
```

Figure 16: Monitors used in the scenario of Figure 13.

these parameters. Therefore, the objective now is to determine the time at which the unhealthy condition is reached. To study this circumstance, we proceed as in the previous case, by performing 100 replications with the values for parameters $f1$ and $f2$ indicated above. Figure 17 shows the results obtained using a *box plot* (the central mark in red indicates the median, and the bottom and top edges of the blue box indicate the first and third quartiles, respectively). The whiskers (discontinuous lines drawn in black from the box) extend to the most extreme data points not considered outliers, where the outliers are plotted using the "+" symbol in red).

Table 1: Results obtained taking different values to increase the *CO* level.

Scenario	Initial	Increment	Probability (%)
1	3	0.5 (fixed)	0
2	3	Uniform (0.5, 0.6)	0
3	3	Uniform (0.5, 0.65)	4
4	3	Uniform (0.5, 0.67)	41
5	3	Uniform (0.5, 0.68)	67
6	3	Uniform (0.5, 0.69)	88
7	3	Uniform (0.5, 0.7)	91
8	3	Uniform (0.5, 0.71)	99
9	3	Uniform (0.5, 0.72)	100

Notice that we have chosen a synthetic scenario to illustrate the applicability of our methodology, i.e., we start with a specific value for the *CO* value and we use different distributions to produce the event information for the next 300 minutes. Other arrangements can be considered, by modifying the initial value and/or the distribution used to change the values in credible values throughout the time. Obviously, the interest of using CPNs and CPN Tools lies in the possibility of using the automatic simulator engine and the monitor capabilities to make performance studies and obtain predictive results of the system behavior.

Finally, the model could be enriched by including the actions that should be taken to deal with the situations detected. As an example, for a detected unhealthy situation (*CO* average is greater than 35 mg/m^3 for 1 hour), we could consider as possible actions to start the air conditioner, open some windows, start some fans, etc. These actions could actually be included in the PCPN model as transitions that would be executed in the case that these conditions were satisfied. Furthermore, we could also use real data obtained from sensors in order to simulate a scenario with this pollution information. Thus, with the results obtained, we could predict unhealthy situations and then start the fans or the air conditioners before they occur.

5. Related Work

Over recent years, some collective computational intelligence technologies and tools have emerged in response to the demands for analyzing big data. Among them, the CEP technology is essential for analyzing the complexity of

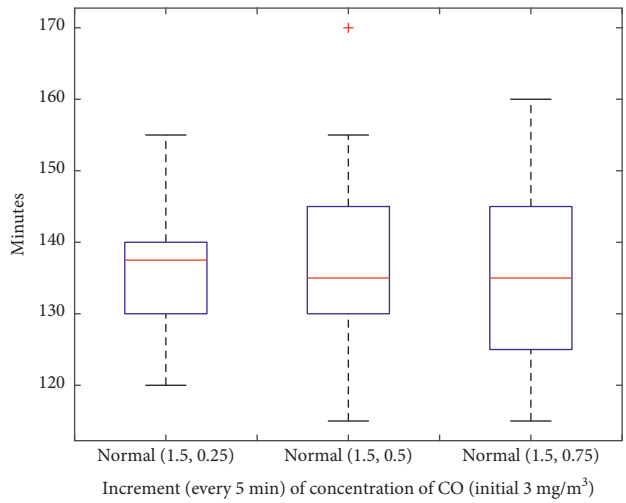


Figure 17: Box plot obtained for different scenarios.

multicriteria that could generate an alert [21], extracting meaningful events. However, the implementation of event pattern conditions can become a handicap for users who are experts on the domain but not in the involved technology. Moreover, conducting the generation of data for semantically validating those patterns is a cumbersome task. To solve these problems, on one hand, some works have proposed the use of Petri nets for conducting the semantic validation of event patterns, while others use Model-Driven Engineering (MDE) techniques for making the CEP technology closer to any user. We propose the combination of both approaches.

Regarding works using the Petri Net formalism, Offel et al. [22] show that formal methods as Petri nets can help in the design and implementation of CEP systems which are underdeveloped, but they are in the process of developing tool support for the envisioned verification of CEP systems. Weidlich et al. [23] used PCPNs with time in order to define a model of event-processing networks. The Event-Processing Network (EPN) architecture is presented, and a general translation of this concept, an example implemented in the ETALIS framework [24], is also presented.

Ahmad et al. [25] describe a methodology modeling CEP using Timed Net Condition Event System (TNCES) [26]. An application to a Manufacturing Line is also presented as an example. NCES is a Petri Net derived formalism based on Condition Event Systems to model discrete event dynamic systems. NCES is extended with time in TNCES, which is based on timed-arc Petri nets [27–29]. Thus, the main difference to our work, other than the Petri Nets formalism used, is that we integrate the PCPN translation into the MEdit4CEP tool, so as to automatically obtain the PCPNs from the event pattern graphical specification created by using this tool.

Other authors like Metzger et al. [30] analyze the CEP systems under the verification perspective applying model checkers. As an example, the authors perform incremental verifications using Petri Nets as models for the Tapaal tool, a bounded model checker. The approach used in this work to deal with the state explosion problem is to gradually increase the size of the model, which is a different approach to analyze CEP systems instead of using the quantitative analysis. Cugola and Margara [31] have defined the TESLA language, which is a complex event specification language, based on a metric temporal logic. TESLA is a highly expressive and flexible language in a rigorous framework, offering content and temporal filters, negations, timers, aggregates, and fully customizable policies for event selection and consumption. Ericsson et al. [32] have defined a prototype tool REX, with support for specifying both CEP systems and correctness properties in a high-level graphical language. CEP applications are then transformed into timed automata, and the UPPAAL tool [33] is used for automatic verification. Agrawal et al. [34, 35] have also defined a timed automata formalization of complex event systems. They present the Sase+pattern language, which defines a precise semantics in terms of timed automata with similar results to the work introduced in TESLA. Cugola and Margara have also proposed CAVE tool [36] with the purpose of assisting domain experts in the definition of a set of reliable rules for a CEP application. In particular, this tool analyzes the behavior of a CEP application transforming the property checking rules into basic constraints solving problems. Additionally, they have also presented a survey [37] of the existing Information Flow Processing (IFP) systems, including CEP systems, activa databases, etc. They show the different approaches and mechanisms adopted in these IFP systems to deal with the event flow processing.

Rewriting logic and the Maude language [38] have been also used to specify and analyze CEP systems. Burgueño et al. [39] propose a framework for the specification of CEP applications, allowing developers to formally analyze and prove properties of their CEP programs. An encoding of CEP concepts and mechanisms to Maude is provided, and several analysis are presented, both covering the static properties of the CEP patterns and the statistical simulation of such systems. Garcí-a-López et al. [40] have implemented the CEPA tool for the transformation of CEP programs to Abstract Syntax Trees (AST) capturing the pattern dependencies, with the goal to check and correct two particular

Table 2: Modeling/analysis approaches based on formal methods.

Approach	Objective	Underlying formalism and tool	Modeling	Model validation	Transformation
MEdit4CEP-CPN [6]	Semantic and quantitative analysis graphically supported by CPN Tools	Colored Petri nets & CPN Tools	Graphical	Syntactic analysis of graphical models	Automatic to EPL and PCPN
CEP in Maude [39]	General formal analysis script supported by Maude	Rewriting logic and Maude	Textual	Syntactic analysis	Manual
CEPA [40]	Rule acyclicity and race condition checking	Graph theory	Textual	Syntactic analysis of textual models	Automatic to AST
REX [32]	Property verification	Timed automata	Graphical	Using UPPAAL	Automatic to timed automata
CAVE [36]	Property verification and performance analysis	Ad hoc CEP language	Textual	Constraint solver	Automatic to configurations

properties of CEP systems: rule acyclicity and rule race conditions.

Concerning the quantitative analysis of complex events, Tendick et al. [41] focus on the use of statistical methods for the CEP technology for making decisions in real time, providing additionally a comparison of computing techniques in widespread use for real-time data. Other approaches like Rajsiri et al.'s work [42] and the one presented here focus on studios via simulations. Rajsiri et al. present a business process editor and simulator developed on the basis of an event-driven business process modeling approach using the BPMN 2.0 formalism. This work dealt with the problem of the business processes simulation taking an even-driven perspective into account to observe the system behavior; however, users cannot automatically replicate scenarios. In addition, there are also several tools for performance evaluation of CEP applications by simulations, for instance, CEPSim [43] is a simulator for CEP and Stream Processing (SP) systems in cloud environments which allows us to analyze the performance and scalability of user-defined queries and to evaluate the effects of various query processing strategies. Mendes et al. [44] have developed FINCoS, which is a set of benchmarking tools for load generation and performance measuring of event-processing systems, so as to make performance evaluation on CEP platforms independently on their structural differences or the workload employed. Along the same lines, Li and Berry [45] have also developed a benchmark of complex event-processing systems focusing on complex event-processing functional behaviours: filtering, transformation, and event pattern detection. They also show the factors that influence performance measurements.

As a summary, Table 2 shows a comparison of our proposal (MEdit4CEP-CPN) with the most representative mentioned modeling/analysis works based on formal methods.

It also deserves special mention the CEP engine proposed by Cugola and Margara [46], T-REX, based on the TESLA language that combines expressiveness and efficiency. T-REX middleware provides an efficient event detection algorithm based on automata to interpret TESLA rules. This work could be used in conjunction to our proposal through the inclusion of additional transformations to generate TESLA rules.

Regarding to works considering metamodels of PNs, Gomez et al. [47] proposed a metamodel for PNs in the domain of *biological data processing*. The models conforming to this metamodel are then transformed into the XML code executable by CPN Tools. This work shows some limitations with respect to our proposal. Among them, PN modeling is close to CPN Tools concepts, and a tree model editor is used to produce the models. In addition, a CPN model can only have one page, and priorities are not considered. Additionally, Westergaard et al. [48] implemented Access/CPN, a framework providing CPN Tools with two interfaces. One is written in Standard Markup Language, which is useful for analysis methods. The other interface is written in Java and provides an object-oriented representation of CPN models, whose object model (metamodel) is implemented by using Eclipse Modeling Framework (EMF). However, the latest version released is not actually up-to-date and, although the latest version available from Subversion (https://svn.win.tue.nl/repos/cpntools/AccessCPN/trunk/) has better support for 4.0 features of CPN Tools, it is still not complete, as stated by Westergaard. In addition, Petri Net modeling is addressed by using a tree model editor (not a graphical one with nodes and links), as in the work by Gomez et al. [47].

6. Conclusions and Future Work

In this paper, we have illustrated the use of the MEdit4CEP-CPN approach for the complex event analysis through a case study based on the sick building syndrome. The event patterns have been graphically modeled with MEdit4CEP-CPN and then automatically transformed into both EPL and CPN code. Additionally, CPN Tools have been used to make quantitative analysis of events produced for this case study. Given the flexibility provided by MEdit4CEP-CPN, this analysis could be applied to other cutting-edge real-world case studies, such as eHealth [49], robotic [50] and mobile edge, and cloud computing applications [51].

As shown in the related work, there are many works using CPNs to model CEP-based languages, but to the best of our knowledge, they do not provide end users with an all-in-one graphical tool with the following goals: (1) modeling CEP domains and event patterns in a user-friendly way by dragging and dropping elements on a canvas, (2) validating the pattern syntax, (3) automatically transforming the graphical patterns

into a CPN model, (4) automatically transforming the CPN model to the XML code executable by CPN Tools and validating the pattern semantics, (5) automatically generating the Esper EPL code and deploying it in a particular event-based system, and (6) providing a quantitative analysis of complex events through the CPN Tools executable model automatically generated by the tool. Let us observe that the MEdit4CEP-CPN model-driven approach presented in this paper provides support for all of these functionalities.

As future work, we plan to add additional features and functionalities to MEdit4CEP-CPN, such as further EPL operators and new transformation techniques. Since the use of CPN Tools requires some knowledge from users in order to conduct the quantitative analysis, at least for modifying the initial marking of the produced CPN model and to execute the simulations to obtain the results, we intend to alleviate this problem by enriching our graphical model for event pattern design. This will make it possible to set the initial conditions (event flow) at design time and adding the option to automatically execute the produced CPN. The obtained output would then be transformed into the corresponding complex events in the output flow.

Data Availability

The obtained PCPN model and simulation data used to support the findings of this study have been deposited in the Mendeley repository (DOI: 10.17632/kfrkyzdxnv.1).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work has been partially supported by the Spanish MINECO/FEDER projects TIN2015-65845-C3-2-R, TIN2015-65845-C3-3-R, and TIN2016-81978-REDT. Boubeta-Puig would like to thank the Real-Time and Concurrent Systems Research Group for their hospitality when visiting them at the University of Castilla-La Mancha, Spain, where part of this work was developed.

References

[1] D. Luckham, *Event Processing for Business: Organizing the Real-Time Enterprise*, John Wiley & Sons, New Jersey, NJ, USA, 2012.

[2] F. Bry, M. Eckert, O. Etzion, J. Riecke, and P. Adrian, "Event processing languages. Tutorial in DEBS 2009," in *Proceedings of the 3rd ACM International Conference on Distributed Event-Based Systems*, Nashville, TN, USA, July 2009.

[3] J. Boubeta-Puig, G. Ortiz, and I. Medina-Bulo, "MEdit4CEP: a model-driven solution for real-time decision making in SOA 2.0," *Knowledge-Based Systems*, vol. 89, pp. 97–112, 2015.

[4] EsperTech, "Esper—complex event processing," February 2019, http://www.espertech.com/esper/.

[5] W. M. P. van der Aalst and C. Stahl, *Modeling Business Processes—A Petri Net-Oriented Approach. Cooperative Information Systems Series*, MIT Press, Cambridge, MA, USA, 2011.

[6] J. Boubeta-Puig, G. Díaz, H. Macià, V. Valero, and G. Ortiz, "MEdit4CEP-CPN: an approach for complex event processing modeling by prioritized colored Petri nets," *Information Systems*, vol. 81, pp. 267–289, 2019.

[7] H. Macia, V. Valero, G. Diaz, J. Boubeta-Puig, and G. Ortiz, "Complex event processing modeling by prioritized colored Petri nets," *IEEE Access*, vol. 4, pp. 7425–7439, 2016.

[8] M. Westergaard and H. M. W. Verbeek, "CPN tools homepage," February 2019, http://www.cpntools.org/.

[9] A. Adi, D. Botzer, N. Gil, and S. Guy, "Complex event processing for financial services," in *Proceedings of the 2006 IEEE Services Computing Workshops*, pp. 7–12, IEEE, Chicago, IL, USA, September 2006.

[10] A. Buchmann, H. C. Pfohl, S. Appel et al., "Event-driven services: integrating production, logistics and transportation," in *Service-Oriented Computing*, E. Michael Maximilien, G. Rossi, S.-T. Yuan, H. Ludwig, and M. Fantinato, Eds., vol. 6568, pp. 237–241, Springer, Berlin, Germany, 2011.

[11] R. Gad, M. Kappes, J. Boubeta-Puig, and I. Medina-Bulo, "Employing the CEP paradigm for network analysis and surveillance," in *Proceedings of the Ninth Advanced International Conference on Telecommunications*, pp. 204–210, IARIA, Rome, Italy, June 2013.

[12] A. Garcia-de-Prado, G. Ortiz, and J. Boubeta-Puig, "COLLECT: COLLaborativE ConText-aware service oriented architecture for intelligent decision-making in the Internet of Things," *Expert Systems with Applications*, vol. 85, pp. 231–248, 2017.

[13] A. Garcia-de Prado, G. O. Ortiz, J. Boubeta-Puig, and D. Corral-Plaza, "Air4People: a smart air quality monitoring and context-aware notification system," *Journal of Universal Computer Science*, vol. 24, no. 7, pp. 846–863, 2018.

[14] W. Wang, H. Yang, Y. Zhang, and J. Xu, "IoT-enabled real-time energy efficiency optimisation method for energy-intensive manufacturing enterprises," *International Journal of Computer Integrated Manufacturing*, vol. 31, no. 4-5, pp. 362–379, 2018.

[15] M. E. Cambronero, H. Macia, V. Valero, and L. Orozco-Barbosa, "Modeling and analysis of the 1-wire communication protocol using timed colored Petri nets," *IEEE Access*, vol. 6, pp. 27356–27372, 2018.

[16] K. Jensen and L. M. Kristensen, *Coloured Petri Nets: Modelling and Validation of Concurrent Systems*, Springer Publishing Company, Berlin, Germany, 1st edition, 2009.

[17] Eclipse Foundation, "Epsilon," June 2019, https://www.eclipse.org/epsilon/.

[18] D. S. Kolovos, A. García-Domínguez, L. M. Rose, and R. F. Paige, "Eugenia: towards disciplined and automated development of GMF-based graphical model editors," *Software & Systems Modeling*, vol. 16, no. 1, pp. 229–255, 2017.

[19] World Health Organization, *Regional Office for Europe*, WHO, Geneva, Switzerland, February 2019, http://www.euro.who.int/en/home.

[20] World Health Organization, *Regional Office for Europe*, WHO Guidelines for Indoor Air Quality: Selected Pollutants, Geneva, Switzerland, February 2019, https://apps.who.int/iris/bitstream/handle/10665/260127/9789289002134-eng.pdf?sequence=1&isAllowed=y.

[21] N. Bessis and F. Xhafa, *Next Generation Data Technologies for Collective Computational Intelligence*, Springer, Berlin, Germany, 2011.

[22] M. Offel, H. van der Aa, and M. Weidlich, "Towards net-based formal methods for complex event processing," in *Proceedings*

of the Conference "Lernen, Wissen, Daten, Analysen", LWDA 2018, pp. 281–284, Mannheim, Germany, August, 2018.

[23] M. Weidlich, M. Jan, and A. Gal, "Net-based analysis of event processing networks—the fast flower delivery case," in *Application and Theory of Petri Nets and Concurrency, LNCS*, J. M. Colom and J. Desel, Eds., vol. 7927, pp. 270–290, Springer, Berlin, Germany, 2013.

[24] D. Anicic and P. Fodor, "Event-driven transaction logic inference system," February 2019, https://code.google.com/archive/p/etalis/.

[25] W. Ahmad, A. Lobov, L. Jose, and M. Lastra, "Formal modelling of complex event processing: a generic algorithm and its application to a manufacturing line," in *Proceedings of the IEEE 10th International Conference on Industrial Informatics, INDIN*, pp. 380–385, Beijing, China, July 2012.

[26] M. Rausch and H. M. Hanisch, "Net condition/event systems with multiple condition outputs," in *Proceedings of the Emerging Technologies and Factory Automation, ETFA'95*, pp. 592–600, Paris, France, October 1995.

[27] D. de Frutos, V. Valero, and O. Marroquín, "Decidability of properties of timed-arc Petri nets," in *Application and Theory of Petri Nets, LNCS*, M. Nielsen and D. Simpson, Eds., vol. 1825, pp. 187–206, Springer, Berlin, Germany, 2000.

[28] H.-M. Hanisch, "Analysis of place/transition nets with timed-arcs and its application to batch process control," in *Application and Theory of Petri Nets, LNCS*, M. Ajmone Marsan, Ed., vol. 691, pp. 282–299, Springer, Berlin, Germany, 1993.

[29] V. Valero, D. de-Frutos, and F. Cuartero, "On non-decidability of reachability for timed-arc Petri nets. Petri nets and performance models," in *Proceedings 8th International Workshop on Petri Nets and Performance Models*, pp. 188–197, IEEE Computer Society Press, Zaragoza, Spain, September 1999.

[30] A. Metzger, C. Reinartz, and K. Pohl, "Incremental verification of complex event processing applications for system monitoring," in *Proceedings of the 2018 44th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, pp. 293–297, Prague, Czech Republic, August 2018.

[31] G. Cugola and A. Margara, "TESLA: a formally defined event specification language," in *Proceedings of the 4th ACM International Conference on Distributed Event-Based Systems, DEBS '10*, pp. 50–61, ACM, Cambridge, UK, July 2010.

[32] A. M. Ericsson, P. Pettersson, M. Berndtsson, and M. Seiriö, "Seamless formal verification of complex event processing applications," in *Proceedings of the 2007 Inaugural International Conference on Distributed Event-Based Systems, DEBS*, pp. 50–61, Toronto, ON, Canada, June 2007.

[33] K. G. Larsen, P. Pettersson, and Y. Wang, "UPPAAL in a nutshell," *International Journal on Software Tools for Technology Transfer*, vol. 1, no. 1-2, pp. 134–152, 1997.

[34] J. Agrawal, Y. Diao, D. Gyllstrom, and N. Immerman, "Efficient pattern matching over event streams," in *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, pp. 147–160, ACM, Vancouver, Canada, June 2008.

[35] D. Gyllstrom, J. Agrawal, Y. Diao, and N. Immerman, "On supporting kleene closure over event streams," in *Proceedings of the 24th International Conference on Data Engineering (ICDE)*, pp. 1391–1393, Cancún, MX, USA, April 2008.

[36] G. Cugola, A. Margara, M. Pezzè, and M. Pradella, "Efficient analysis of event processing applications," in *Proceedings of the 9th ACM International Conference on Distributed Event-Based Systems, DEBS '15*, pp. 10–21, Oslo, Norway, June-July 2015.

[37] G. Cugola and A. Margara, "Processing flows of information: from data stream to complex event processing," *ACM Computing Surveys*, vol. 44, no. 3, pp. 1–62, 2012.

[38] M. Clavel, F. Durán, S. Eker et al., "All about Maude—a high-performance logical framework, how to specify, program and verify systems in rewriting logic," in *Lecture Notes in Computer Science*, Vol. 4350, Springer, Berlin, Germany, 2007.

[39] L. Burgueno, J. Boubeta-Puig, and A. Vallecillo, "Formalizing complex event processing systems in Maude," *IEEE Access*, vol. 6, pp. 23222–23241, 2018.

[40] A. García-López, L. Burgueño, and A. Vallecillo, "Static analysis of complex event processing programs," in *Proceedings of the MODELS 2018 Workshops*, vol. 14, pp. 498–502, Copenhagen, Denmark, October 2018.

[41] P. H. Tendick, L. Denby, and W.-H. Ju, "Statistical methods for complex event processing and real time decision making," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 8, no. 1, pp. 5–26, 2016.

[42] V. Rajsiri, N. Fleury, C. Graham, and J.-P. Lorré, "Event-based business process editor and simulator," in *Business Process Management Workshops—BPM 2010 International Workshops and Education Track*, M. zur Muehlen and J. Su, Eds., Springer, Berlin, Germany, 2011.

[43] W. A. Higashino, M. A. M. Capretz, and L. F. Bittencourt, "CEPSim: modelling and simulation of complex event processing systems in cloud environments," *Future Generation Computer Systems*, vol. 65, pp. 122–139, 2016.

[44] M. R. N. Mendes, B. Pedro, and P. Marques, "FINCoS: benchmark tools for event processing systems," in *Proceedings of the ACM/SPEC on International conference on Performance Engineering—ICPE'13*, pp. 431-432, Prague, Czech Republic, August 2013.

[45] C. Li and R. Berry, "CEPBen: a benchmark for complex event processing systems," in *Performance Characterization and Benchmarking*, R. Nambiar and M. Poess, Eds., pp. 125–142, Springer International Publishing, Cham, Switzerland, 2014.

[46] G. Cugola and A. Margara, "Complex event processing with T-REX," *Journal of Systems and Software*, vol. 85, no. 8, pp. 1709–1728, 2012.

[47] A. Gomez, A. Boronat, J. A. Carsi, I. Ramos, C. Taubner, and S. Eckstein, "Biological data processing using model driven engineering," *IEEE Latin America Transactions*, vol. 6, no. 4, pp. 324–331, 2008.

[48] M. Westergaard and L. Michael Kristensen, "The Access/CPN framework: a tool for interacting with the CPN Tools simulator," in *Applications and Theory of Petri Nets*, G. Franceschinis and K. Wolf, Eds., pp. 313–322, Springer, Berlin, Germany, 2009.

[49] A. Camacho, M. G. Merayo, and M. Núñez, "Collective intelligence and databases in eHealth: a survey," *Journal of Intelligent & Fuzzy Systems*, vol. 32, no. 2, pp. 1485–1496, 2017.

[50] A. Rutle, J. Backer, K. Foldøy, and R. T. Bye, "Commonlang: a DSL for defining robot tasks," in *Proceedings of MODELS 2018 Workshops: ModComp, MRT, OCL, FlexMDE, EXE, COMMitMDE, MDETools, GEMOC, MORSE, MDE4IoT, MDEbug, MoDeVVa, ME, MULTI, HuFaMo, AMMoRe, PAINS Co-Located with ACM/IEEE 21st International Conference on Model Driven Engineering Languages and Systems (MODELS 2018)*, pp. 433–442, Copenhagen, Denmark, October 2018.

[51] G. Orsini, D. Bade, and W. Lamersdorf, "CloudAware: empowering context-aware self-adaptation for mobile applications," *Transactions on Emerging Telecommunications Technologies*, vol. 29, no. 4, e3210 pages, 2018.

Hindawi
Scientific Programming
Volume 2019, Article ID 8043905, 18 pages
https://doi.org/10.1155/2019/8043905

Review Article

Recommendation and Classification Systems: A Systematic Mapping Study

J. G. Enríquez,[1] **L. Morales-Trujillo**,[1] **Fernando Calle-Alonso**,[2] **F. J. Domínguez-Mayo**,[1] **and J. M. Lucas-Rodríguez**[3]

[1]*Computer Languages and Systems Department, University of Seville, Avd. Reina Mercedes s/n, 41012 Seville, Spain*
[2]*Statistics and Operational Research Area, University of Malaga, Bulevar Louis Pasteur 31, 29010 Malaga, Spain*
[3]*Servinform S.A., Calle Manufactura, 5, 41927 Mairena del Aljarafe, Spain*

Correspondence should be addressed to J. G. Enríquez; jose.gonzalez@iwt2.org

Received 15 February 2019; Revised 14 May 2019; Accepted 12 June 2019; Published 27 June 2019

Guest Editor: Luis Iribarne

Today, recommendation algorithms are widely used by companies in multiple sectors with the aim of increasing their profits or offering a more specialized service to their customers. Moreover, there are countless applications in which classification algorithms are used, seeking to find patterns that are difficult for people to detect or whose detection cost is very high. Sometimes, it is necessary to use a mixture of both algorithms to give an optimal solution to a problem. This is the case of the ADAGIO, a R&D project that combines machine learning (ML) strategies from heterogeneous data sources to generate valuable knowledge based on the available open data. In order to support the ADAGIO project requirements, the main objective of this paper is to provide a clear vision of the existing classification and recommendation ML systems to help researchers and practitioners to choose the best option. To achieve this goal, this work presents a systematic review applied in two contexts: scientific and industrial. More than a thousand papers have been analyzed resulting in 80 primary studies. Conclusions show that the combination of these two algorithms (classification and recommendation) is not very used in practice. In fact, the validation presented for both cases is very scarce in the industrial environment. From the point of view of software development life cycle, this review also shows that the work being done in the ML (for classification and recommendation) research and industrial environment is far from earlier stages such as business requirements and analysis. This makes it very difficult to find efficient and effective solutions that support real business needs from an early stage. It is therefore that the article suggests the development of new ML research lines to facilitate its application in the different domains.

1. Introduction

The great growth in the amount of data and information that can be accessed (the known Big Data), coupled with government collaboration to provide open information (Open Data), makes companies very interested in this issue. One of the biggest problems in this area is that this information is not found in one single place, not even in a common interpretation format. Therefore, it is necessary to create solutions that collect these dispersed data and apply a specific treatment so that they can be offered to their customers.

The collection of dispersed information and its unification in order to be able to work with it would open a new market niche, a new business unit, considering the possibility of generating valuable data automatically. In addition, it would increase independence when making decisions or solving problems without having to resort to an expert in business management.

The ADAGIO project was born in this context. It is a R&D project that combines Big Data and machine learning (ML) strategies for the treatment of geolocated data extracted from heterogeneous data sources. It enables the aggregation, consolidation, and normalization of data from

different semantic fields obtained from the sources mentioned before. Its purpose is to allow reconciled information to be consulted using specific variables, thus facilitating the generation of knowledge.

The application of classification and recommendation systems in this project is of great interest for the interrelation and periodic consolidation of the data process so that the system develops capabilities for transformation, interrelation, and integration of data through supervised learning. In addition, these systems provide a great value for the management of queries, to enhance the performance of queries by users in a language as natural and high level as possible. The fact that the user obtains good results during the searches in the ADAGIO platform is one of the main objectives of the project. In order to improve the user's experience, suggestions are proposed during the phase of filling the search parameters. For this phase, the collaboration of the system users will also be required, evaluating the results of the searches according to their quality and precision.

This study has been performed to facilitate researchers and practitioners the task of choosing the most appropriate system, technology, or algorithm to include in the ADAGIO project for satisfying their requirements. In this sense, this paper presents a systematic mapping study (SMS) that analyzes the current state of the art of the recommendation and classification systems and how they work together. Then, from the point of view of the software development life cycle, this review also shows that the work being done in the ML (for classification and recommendation) research and industrial environment is far from earlier stages such as business requirements and analysis. This makes it very difficult to find efficient and effective solutions that support real business needs from an early stage. Then, this paper suggests the development of new ML research lines to facilitate its application in the different domains.

This paper is organized as follows. Section 2 describes the closest related work to our proposal; Section 3 details the selected method to carry out the SMS; Sections 4 to 8 illustrate the execution of the different phases of the SMS; and finally, Section 9 summarizes the conclusions obtained from the study and presents a set of future work.

2. Related Work

Recommendation and classification systems are acquiring much interest within the scientific community. In this section, the closest related works to the research proposed in this article are presented.

Jaysri et al. [1] presented a complete review of the recommendation systems, focusing on the collaborative filtering. It shows different algorithms based on this filtering for both the user profile and the product characteristics. In addition, it demonstrates several classification methods that may be part of the input for recommendation systems. Ekstrand et al. [2] presented a general overview and focused on the field of recommendation systems. Their purpose was to learn more about the current development of recommendation methods, specially systems making use of the collaborative filtering.

Obtaining a research perspective on how to make decisions when choosing algorithms to propose recommendations can be found in the paper presented by Gunawardana and Shani [3]. It criticizes the use of online methods, which can offer measures to choose recommendation algorithms, and determines as a crucial element the use of offline tools to obtain these measures. In addition, it discards the use of traditional metrics to make the algorithm choice and reviews the proper elaboration of experiments to carry it out. To do this, the authors perform an analysis of important tasks of the recommendation systems and classify a set of appropriate and well-known assessment measures for each task.

Poussevin et al. [4] exposed the challenge of considering the preferences of users when recommending. The authors analyzed a combination of recommendation systems and classifiers that highlight words that indicate a gap between users' expectations and their actual experience. They conclude that traditional recommendation systems analyze the past classifications; that is, they consider the users' preferences history, while the recommendation systems that analyze the opinion classifications consider the existing evaluations at that moment.

Within the scope of ML, there has been an increase in the interest of the research community, being the subject of many papers. Some of the proposals use lexical classifiers to detect possible feelings using content-based recommendations [5]. Other authors have focused on more traditional branches of ML, using well-known and proven statistical methods such as logistic regression, the Pearson correlation coefficient, or the application of the naive Bayes theorem based on probability, among others [6]. The authors of this paper focused on making extensions of these methods to solve problems inherent in recommendation systems such as cold start or scalability. The cold start [7] is a typical problem since the beginning of the recommendation systems because when a system does not have enough data, precision cannot be assured when recommending. This is a problem that gets worse at the beginning of the implementation of a system when data are not available. Scalability becomes a quite difficult task due to the increase of information in recent years and the amount of data that systems must manage. Recommendation systems, both product and user-based, affect performance and accuracy when these amounts of data are very large. The work presented by Ghazanfar and Prügel-Bennett [8] has been also focused on this problem, generally for the user-based recommendation, which is the most used.

Alternative interesting related work focused in the use of ML is the survey in sentiment classification presented by Hailong et al. [9]. In this work, the authors also provide a comparative study of the techniques found, concluding that supervised ML present a higher accuracy, while lexicon-based methods are likewise competitive because they require less effort and they are not sensitive to the quantity and quality of the training dataset. The survey presented by Mu [10] delivers a review of deep learning-based recommender systems. The authors conclude this work summarizing a set

of future research lines such as cross domain, scalability, explainability, or deep composite model-based recommender systems, among others.

The paper presented by Portugal et al. [11] presents a systematic review of the use of ML in recommender systems. The authors analyzed 121 primary studies classified in different categories: content-based and neighbor-based of content-based filtering, neighborhood-based and model-based of collaborative filtering, and hybrid filtering. This work helps developers to recognize the algorithms, their types, and trends in the use of specific algorithms. It also offers current-type evaluation metrics and categorizes the algorithms based on these metrics. Ouhbi et al. [12] proposed a deep learning-based recommender system to overcome some limitations of existing approaches. In the related work section of this paper, the authors describe a small state of the art of deep learning-based recommender systems, detailing the method, approach, metric, dataset, advantages, and disadvantages of seven proposals.

Zhang et al. [13] delivered a wide review of deep learning-based recommender systems, proposing a classification and highlighting a group of the most influential. The authors debate the pros and cons of using deep learning techniques for recommendation tasks. Additionally, some of the most pressing open problems and promising future extensions are detailed.

In summary, the literature review presented different topics, which may come close to the objective pursued. But there are several differences between these papers and the one presented in this work: (i) the review process: unlike the rest of the papers, this research presents a systematic and rigorous process, ensuring the quality of the results obtained; (ii) the context of application: usually reviews are carried out on the scientific literature; in this case, this research also presents a review on the industrial scope, analyzing the main existing solutions to the problem; and (iii) the scope of application: in this systematic review, the state of the art of the classification and recommendation systems is presented working together, something that in the related works already mentioned is not carried out or it is done independently for classification or recommendation.

3. Methodology

A systematic literature review is an effective way of knowing the state of the art of a subject. This procedure ensures a certain level of quality of information and has the support of the research community. The monitoring of a systematic and guided process guarantees reliable and interesting results and facilitates the work of gathering information.

The review presented in this paper is placed within the context of the recommendation and classification systems from two perspectives: scientific and industrial.

When carrying out a systematic literature review (SLR), the main methodology to be considered is the one presented by Kitchenham and Charters [14]. This is one of the most widely accepted methods in the area of software engineering. It offers a way of performing a SLR consisting in three phases: planning and conducting the review and reporting of results. However, instead of performing a deep review of the papers comparing them, which is the main goal of a SLR, this study seeks to provide an overview of an interesting topic and to identify the number and type of published-related researches, as well as the related results available. Therefore, the best methodology to be applied is the systematic mapping study (SMS) presented by Petersen et al. [15], a type of the systematic review but with a broader objective. This method will allow identifying the subjects that lack empirical evidence and which are necessary to carry out more empirical studies. SMSs show many similarities with respect to the SLRs. As possible to see in activity diagram of Figure 1, this method stablishes a set of five steps, where each of them produces an output. These steps are as follows:

(i) *Definition of the Research Questions.* Formulation of the research questions (RQs) that will guide the work.

(ii) *Conduct Search.* The search is normally executed in different digital libraries and based on some keywords extracted from the RQs.

(iii) *Screening of Papers.* Applying the inclusion and exclusion criteria with the aim of selecting the most relevant and close papers to the topic of the research.

(iv) *Keywording Using Abstract*. Building of the classification scheme, where all the primary papers selected in the previous phase will be categorized.

(v) *Data Extraction and Mapping Process*. Data extraction and mapping process based on the results obtained in the keywording activity. This activity will let the researchers to classify which is the state of the art of the topic and to identify gaps and possibilities for future research.

4. Definition of Research Questions

A Research Question (RQ) is the fundamental core of a research project, study, or literature review. Therefore, to know and better understand the existing literature related to the recommendation and classifications systems, it is necessary to formulate a set of research questions. These questions will focus the study, will determine the methodology that will be established, and will guide all the stages of this research. In this sense, the RQs that have been proposed for this SMS are as follows:

(i) *RQ1*. Which recommendation and classification systems have been researched?

(ii) *RQ2*. Which recommendation and classification systems have been used?

(iii) *RQ3*. Which is the nature of the systems found?

(iv) *RQ4*. Which are the objectives pursued in the proposals found?

5. Conduct Search

Before performing the search in the different digital libraries, it is necessary to complete two operations: define the digital

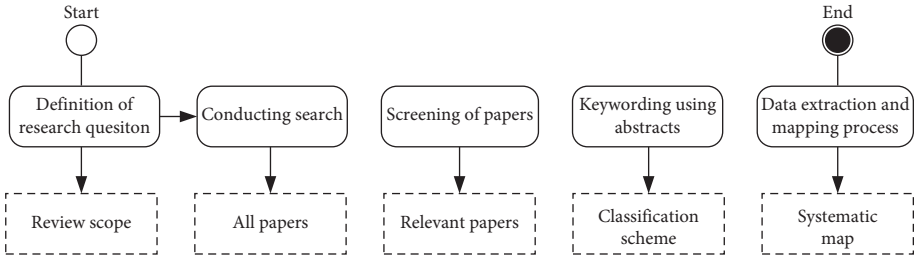


Figure 1: SMS workflow.

libraries where the searches will be executed and establish the keywords that will compose the search strings. Selected digital libraries to carry out the search have been the following: SCOPUS, IEEE Xplore, ACM, and ScienceDirect. In addition, for the industrial scope, the search engines that have been selected are Google, Yahoo, and Bing.

To specify the search, keywords were defined, and it is a fundamental part when creating the queries for each digital library. These keywords were obtained after carrying out an analysis of the field of study to which this research applies, recommendation and classification systems. Table 1 shows the complete set of keywords used, and equation (1) shows the formula applied to these keywords to create the final queries.

Boolean expression of keywords is as follows:

$$E_1 = \left(V_{i=1}^{3} A_i\right) \wedge \left(V_{j=1}^{4} B_j\right) \wedge \left(V_{k=1}^{4} C_k\right). \quad (1)$$

Once all the keywords were defined, the queries were constructed. These queries were different for each digital library, and they had different boundary characteristics, depending on the possibilities of the digital library. Digital libraries have certain limitations when conducting searches. For example, some of them do not allow the use of complete search strings; in others, it is necessary to complement these strings with simple textual searches. For this reason, there is the need to create individual queries for each library and, subsequently, to treat the search results to obtain the same results that could have been obtained using the originally proposed query. Table 2 shows a set of examples for each of digital library.

The search was executed on the title, abstract, and keywords of the papers, except in those digital libraries that did not allow it. In such cases, the search was performed on the complete text. Search strings, metadata of found elements (title, author, and year of publication), and summaries of the documents were stored for each search source. Once the first search was executed, it obtained an initial set of 1,195 potential primary studies.

Table 1: Keyword definition.

A	*B*	*C*
*A*1. Machine learning	*B*1. Recommender system	*C*1. Classifier
*A*2. Deep learning	*B*3. Recommended	*C*2. Classification
*A*3. Neuronal networks	*B*4. Content-based filtering	*C*3. Classified
	*B*5. Collaborative filtering	*C*4. Classification system

Table 2: Example of queries.

Digital library	Query
Scopus	TITLE-ABS-KEY (“machine learning” OR “deep learning” OR “neuronal networks”) AND TITLE-ABS-KEY (“classifier” OR “classification” OR “classified” OR “classification system”) AND TITLE-ABS-KEY (“recommended” OR “recommender system” OR “collaborative filtering” OR “content-based filtering”)
IEEE Xplore	(“Machine learning” OR “deep learning” OR “neuronal networks”) AND (“classifier” OR “classification” OR “classified” OR “classification system”) AND (“recommended” OR “recommender system” OR “collaborative filtering” OR “content-based filtering”)
ACM	acmdlTitle: (“machine learning” “deep learning” “neuronal networks”) OR recordAbstract: (“machine learning” “deep learning” “neuronal networks”) AND (acmdlTitle: (“classifier” “classification” “classified” “classification system”) OR recordAbstract: (“classifier” “classification” “classified” “classification system”)) AND (acmdlTitle: (“recommended” “recommender system” “collaborative filtering” “content-based filtering”) OR recordAbstract: (“recommended” “recommender system” “collaborative filtering” “content-based filtering”))
Science direct	(“Machine learning” OR “deep learning” OR “neuronal networks”) AND (“classifier” OR “classification” OR “classified” OR “classification system”) AND (“recommended” OR “recommender system” OR “collaborative filtering” OR “content-based filtering”)

6. Screening of Papers

There are different metrics to define the quality criteria that make a paper relevant. In this work, in addition to those related to the structure of the papers, the quality assurance criteria defined by those scientific papers found that were classified in the following accepted indexes:

(i) “Journal Citation Report (JCR)” [16] part of the company Thomson Scientific

(ii) The Australian classification created by the “Computing Research and Education Association of Australasia (CORE)” [17]

(iii) The ranking of relevant congresses for the Scientific Information Society of Spain (SCIE) [18], advising the use of the ranking developed by the Italian associations GII and GRIN [19]

In addition, the following inclusion and exclusion criteria were defined for including or being not a publication into the selected primary studies:

(i) *C1, Criterion 1*. The classification of the publication in question must be "Computer Science"

(ii) *C2, Criterion 2*. Written in English

(iii) *C3, Criterion 3*. The research must be related to the classification and recommendation of data using machine learning systems

(iv) *C4, Criterion 4*. Searches cannot be repeated. Multiple appearances must be eliminated

(v) *C5, Criterion 5*. As mentioned above, papers must be classified into the JCR or SCIE rakings

(vi) *C6, Criterion 6*. The reading of the abstract must fit with the dealt topic

Finally, some recommendations from experts in the subject dealt with in this SMS have also been considered. If these studies were not found after the execution of the different searches, they were included in the final selection of primary studies.

Once defined the quality and inclusion and exclusion criteria, the screening of the papers was performed. According to the C1 of inclusion/exclusion of papers which scope is related to "Computer science," a total of 923 results were obtained, having discarded 272 papers that did not meet this criterion. C2 was applied to the 923 papers obtained from C1 resulting on 909 papers. To the results obtained from C2, C3 criterion was applied leaving a total of 432 results. Once C4 was applied, a total of 96 papers were removed remaining 336. A total of 259 papers was the result of applying C5. The last filter, C6, was applied resulting on 99 papers considering that 160 of the removed ones did not fit the topic of this research. Finally, repeated papers were removed. This process ended up removing duplicated entries between the different digital libraries.

The result of applying all the quality and inclusion and exclusion criteria was a total of 80 primary studies which will be categorized into the classification schema. The number of papers found corresponds (roughly) to 6% of the results found in the first search. Table 3 shows the primary studies selected.

Figure 2 shows the list of keywords discovered in the different primary studies. In this figure, the keywords are classified based on the total number of matches found between all these primary studies.

Figure 3 depicts the complete process of selecting primary studies. It shows the search procedure for each digital library and the results after the application of each quality and inclusion and exclusion criteria.

By the same token, the process carried out previously was executed for the industrial scope for detecting and

Table 3: Selected primary studies.

Title	Reference
Building accurate and practical recommender system algorithms using machine learning classifier and collaborative filtering	[20]
DGA botnet detection using collaborative filtering and density-based clustering	[21]
A multistage collaborative filtering method for fall detection	[22]
Analysis and performance of collaborative filtering and classification algorithms	[1]
Extracting a vocabulary of surprise by collaborative filtering mixture and analysis of feelings	[4]
Content based filtering in online social network using inference algorithm	[23]
Building switching hybrid recommender system using machine learning classifiers and collaborative filtering	[8]
Imputation-boosted collaborative filtering using machine learning classifiers	[24]
CRISP-an interruption management algorithm based on collaborative filtering	[25]
A credit scoring model based on collaborative filtering	[26]
Collaborative filtering recommender systems	[2]
An improved switching hybrid recommender system using naive Bayes classifier and collaborative filtering	[6]
Tweet modeling with LSTM recurrent neural networks for hashtag recommendation	[27]
A two-stage cross-domain recommendation for cold start problem in cyber-physical systems	[28]
ELM based imputation-boosted proactive recommender systems	[29]
Twitter-user recommender system using tweets: a content-based approach	[30]
A personalized time-bound activity recommendation system	[31]
Automated content based short text classification for filtering undesired posts on Facebook	[32]
Shilling attack detection in collaborative recommender systems using a meta learning strategy	[33]
Building a distributed generic recommender using scalable data mining library	[34]
Context-aware movie recommendation based on signal processing and machine learning	[35]
Recommender systems using linear classifiers	[36]
A survey of accuracy evaluation metrics of recommendation tasks	[3]
Incorporating user control into recommender systems based on naive Bayesian classification	[37]
Classification features for attack detection in collaborative recommender systems	[38]
Automatic tag recommendation algorithms for social recommender systems	[39]
Optimizing similar item recommendations in a semi-structured marketplace to maximize conversion	[40]
Capturing knowledge of user preferences: ontologies in recommender systems	[41]
Emotion-based music recommendation using supervised learning	[42]
AWESOME—a data warehouse-based system for adaptive website recommendations	[43]

Table 3: Continued.

Title	Reference
Lexical and syntactic features selection for an adaptive reading recommendation system based on text complexity	[5]
A smart-device news recommendation technology based on the user click behavior	[44]
Recommendation as link prediction in bipartite graphs: A graph kernel-based machine learning approach	[45]
A novel approach towards context based recommendations using support vector machine methodology	[46]
A smartphone-based activity-aware system for music streaming recommendation	[47]
An app usage recommender system: improving prediction accuracy for both warm and cold start users	[48]
Proposing design recommendations for an intelligent recommender system logging stress	[49]
A recommender system based on implicit feedback for selective dissemination of eBooks	[50]
A novel recommender system based on FFT with machine learning for predicting and identifying heart diseases	[51]
An approach to content based recommender systems using decision list based classification with k-DNF rule set	[52]
Probabilistic approach for QoS-aware recommender system for trustworthy web service selection	[53]
Approach to cold-start problem in recommender systems in the context of web-based education	[54]
Context and intention-awareness in POIs recommender systems	[55]
A collaborative filtering-based re-ranking strategy for search in digital libraries	[56]
Learning users' interests by quality classification in market-based recommender systems	[57]
Mobile content recommendation system for re-visiting user using content-based filtering and client-side user profile	[58]
A hybrid collaborative filtering algorithm based on KNN and gradient boosting	[59]
A scalable collaborative filtering algorithm based on localized preference	[60]
Recommended or not recommended? Review classification through opinion extraction	[61]
Meta-feature based data mining service selection and recommendation using machine learning models	[62]
Personalized channel recommendation deep learning from a switch sequence	[63]
Affective labeling in a content-based recommender system for images	[64]
A novel approach towards context sensitive recommendations based on machine learning methodology	[65]
A distance-based approach for action recommendation	[66]
Ranking and classifying attractiveness of photos in folksonomies	[67]
Consequences of variability in classifier performance estimates	[68]

Table 3: Continued.

Title	Reference
Machine learning and lexicon based methods for sentiment classification: a survey	[9]
Machine learning algorithm selection for forecasting behavior of global institutional investors	[69]
Towards rapid interactive machine learning: evaluating tradeoffs of classification without representation	[70]
Towards a method for automatically evolving Bayesian network classifiers	[71]
A machine learning based trust evaluation framework for online social networks	[72]
Automated problem identification: regression vs. classification via evolutionary deep networks	[73]
Empirical evaluation of ranking prediction methods for gene expression data classification	[74]
Inferring contextual preferences using deep auto-encoding	[75]
Automatic recognition of text difficulty from consumers health information	[76]
A hybrid approach for automatic model recommendation	[77]
Learning instance greedily cloning naive Bayes for ranking	[78]
Pairwise-ranking based collaborative recurrent neural networks for clinical event prediction	[79]
Accurate multi-criteria decision making methodology for recommending machine learning algorithm	[80]
A general extensible learning approach for multi-disease recommendations in a telehealth environment	[81]
An efficient recommendation generation using relevant jaccard similarity	[82]
An image-based segmentation recommender using crowdsourcing and transfer learning for skin lesion extraction	[83]
Automatic classification of high resolution land cover using a new data weighting procedure: the combination of k-means clustering algorithm and central tendency measures (KMC–CTM)	[84]
Building a hospital referral expert system with a prediction and optimization-based decision support system algorithm	[85]
Classification techniques on computerized systems to predict and/or to detect apnea: a systematic review	[86]
Identification of category associations using a multilabel classifier	[87]
Making use of associative classifiers in order to alleviate typical drawbacks in recommender systems	[88]
S3Mining: a model-driven engineering approach for supporting novice data miners in selecting suitable classifiers	[89]
The use of machine learning algorithms in recommender systems: a systematic review	[11]

selecting the primary technologies or tools that companies offer. The search engines returned multiple results (Table 4), with a total of 21 proposals remaining were potential candidates.

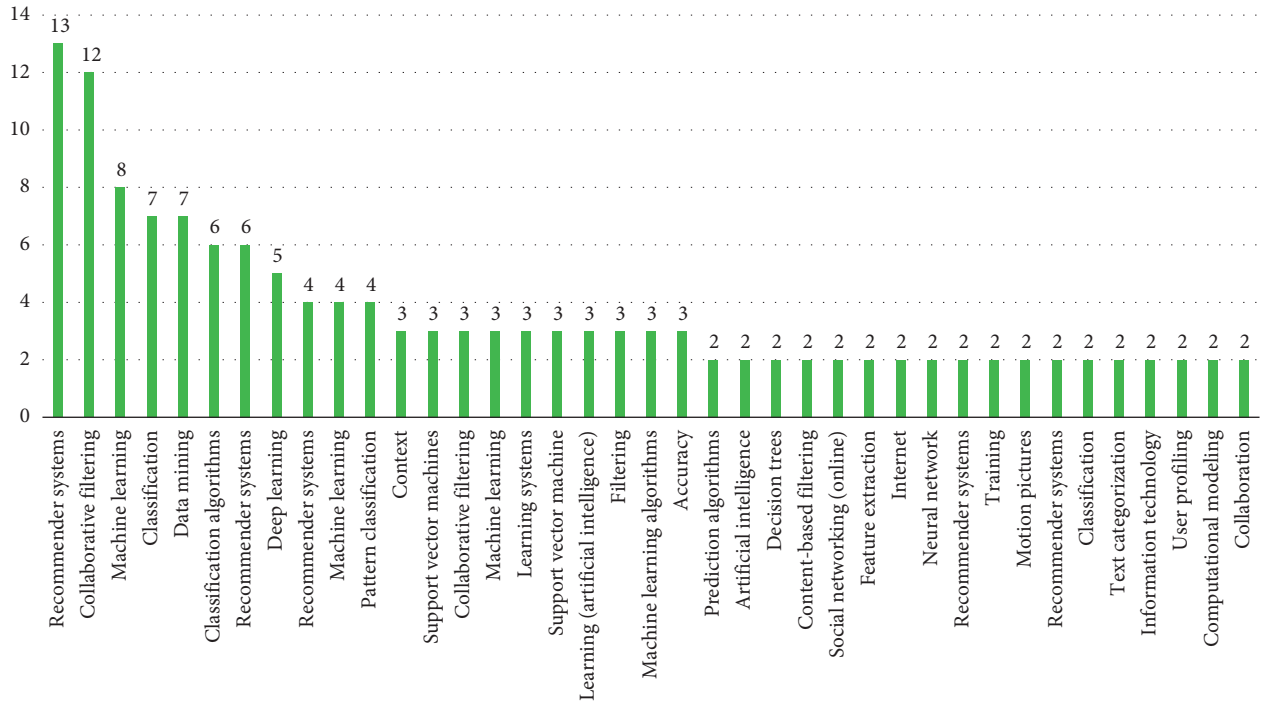


Figure 2: Keyword mapping.

7. Keywording using Abstracts

To create the classification scheme for categorizing the selected primary studies, an attempt was made to answer each of the research questions formulated in the planning phase and, in addition, to identify each of them with a set of features.

Moreover, two complete iterations were carried out to classify all the studies and to verify that all the features that had been found included the content of each study. Table 5 shows and describes the classification scheme defined.

Thereupon, process for the definition of the classification scheme is repeated for the industrial area. Through the answer to the research questions and the extraction of the technologies' features, a classification scheme was defined (Table 6).

8. Data Extraction and Mapping Process

8.1. Scientific Report. This section describes the most important aspects obtained from the information collected. To achieve this purpose, each of the research questions will be answered and validated, showing the data obtained for each of them. It is important to note that some of the features may appear in several studies; therefore, the totals may not always correspond to 100%.

(i) Research Question RQ1 finds the methods, techniques, and/or tools that have been investigated for the classification and recommendation systems. Figure 4 shows that the predominant type of studies is methods, which represent 35.00% of the total of the studies, followed by the complete system studies, with a 23.75%. The rest of studies correspond to algorithms with 20.00%, analysis with a presence of 18.75%, and finally, frameworks with a 6.25% of the total primary studies. From a software development life-cycle perspective (and avoiding methodological discussions), requirements and analysis phases differ from the design phase because it is an earlier stage and closer to the business (or the application model) and is completely technology independent. Then, the found works are contextualized in the technological design phase. No contextualized work was found in early stages (business requirements or analysis).

(ii) Research Question RQ2 seeks to know the validation of the studies found, which may be practical or theoretical, identifying if they are within the scientific or industrial scope. The results obtained (Figure 5) show that all the primary studies were academic focused. Most of them were validated by some way (97.50%), while 10.00% were not validated.

It is important to note that three different groups have been distinguished within the validation category. The experimentation subgroup includes all those studies whose proposal was tested and validated by experimentation with synthetic and real data sources. This group contains most of the results found that were validated, 72.50% of the total. Another important category is the one that validates the proposals by a case study, which represent 13.75%. Only the 5.00% of the primary studies were carried out through surveys, and just one primary study was focused on the industrial context, representing the 1.25% of the total.

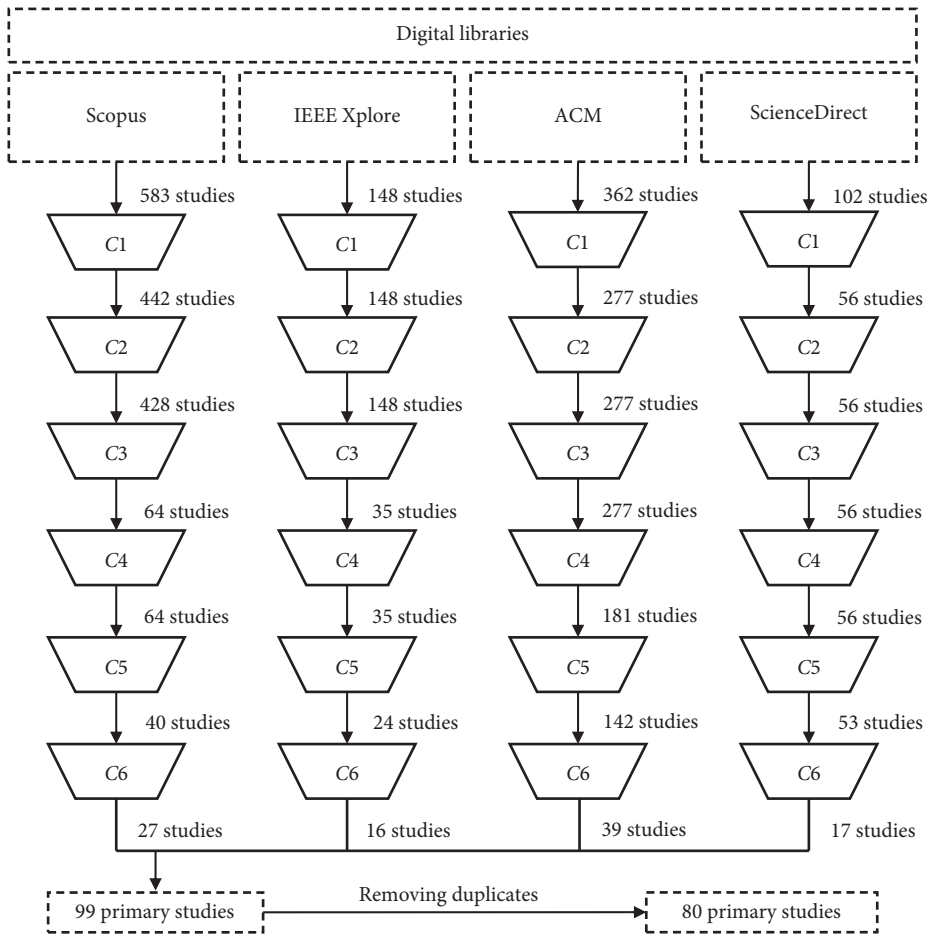


Figure 3: Search flow diagram: scientific field.

(iii) Research Question 3 aims to identify the nature of the methods, techniques, and/or tools about the classification and recommendation systems found in the literature. Figure 6 groups two main categories that contain the whole set of features of the primary studies found: recommendation and classification. Within the recommendation group, content-based and collaborative filtering proposals are very balanced, representing the 36.25% and 38.75%, respectively. Hybrid systems are the worst classified with 17.50% of the papers.

Furthermore, the classification group is described, where both supervised and unsupervised learning features are presented. Two features stand out for their use: naive Bayes to classify according to probabilities with a 28.75% and support vectors, representing the 20.00% of total. Target based and Random Forest are the less used, with a presence of just 1 primary study.

(iv) Research Question RQ4 indicates which are the main points of interest of the research and which areas have been less investigated. This interest is classified into four categories: novelty, analysis, research, and improvement (Figure 7).

The novelty contains those primary studies whose goal is to present something that lacked in the literature, and this category represents 22.50%, with 18 primary studies. Analysis category contains those results that are comparison or study of different existing techniques, and it represents the 7.50% of total. The improvement category represents that 30.00% of the results whose main objective is to improve an existing approach. Finally, the largest category is the research one, were a search on existing or new approaches in the literature is dealt with. It represents the 36.25% of total with 29 primary studies.

At last, it is interesting to analyze other results that are not related to the research questions but with the objective of this document. These results can help to know the evolution of the research of the classification and recommendation systems.

Table 4: Selected primary technologies/tools.

Technology	Reference
Scikit-learn	[90]
Surprise	[91]
LightFM	[92]
Rexy	[93]
PredictionIO	[94]
HapiGER	[95]
LensKit	[96]
SuggestGrid	[97]
SLI Systems Recommender	[98]
AmazonWebService Machine Learning	[99]
Azure ML Studio	[100]
Yusp	[101]
IBM Watson	[102]
Recombee	[103]
Mr. DLib	[104]
Caret	[105]
Shiny	[106]
RandomForest	[107]
KlaR	[108]
CORElearn	[109]
RecommenderLab	[110]

Table 5: Scientific classification scheme.

Research Question	Feature	Description
RQ1	Algorithm	This feature defines if the primary study proposes an algorithm or series of algorithms
	System	This feature defines if the primary study is a software system based on different components
	Framework	This feature defines whether the primary study is based on a framework
	Method	This feature defines if the primary study is a set of procedures to obtain a result
	Analysis	This feature defines if the primary study is a theoretical study based on surveys or systematic reviews among others
RQ2	Validated	This feature defines whether the primary study has been validated with experiments, use cases, or surveys
	Not validated	This feature defines if the primary study has not been validated with experiments, use cases, or surveys
	Academic	This feature defines if the primary study has been validated with some academic case study
	Industrial	This feature defines if the primary study has been validated with some case study in the industry
	Experiment	This feature defines if the primary study has been validated with the elaboration of different experiments
	Use case	This feature defines whether the primary study has been validated with the study of a use case
	Survey	This feature defines if the primary study has been validated with the elaboration of some type of survey

Table 5: Continued.

Research Question	Feature	Description
RQ3	Content based	This feature defines whether the solution proposed by the primary study is based or composed of a recommendation system with a content-based filter
	Collaborative	This feature defines whether the solution proposed by the primary study is based or composed of a recommendation system with a collaborative filter
	Hybrid	This feature defines whether the solution proposed by the primary study is based or composed of a collaborative and content-based filter conjunction
	Graph kernel	This feature defines whether the primary study is based or composed of a graphic classifier
	Naive Bayes	This feature defines whether the primary study is based or composed of a naive probabilistic classifier naive Bayes
	Logistic regression	This feature defines whether the primary study is based or composed of a classifier by logistic regression
	Decision tree	This feature defines whether the primary study is based or composed of a classifier by decision trees
	Lexical	This feature defines whether the primary study is based or composed of a classifier based on textual features
	Based on rules	This feature defines if the primary study is based or composed of a rule-based classifier
	Neural networks	This feature defines if the primary study is based or composed of a classifier based on neural networks
	Clustering	This feature defines if the primary study is based or composed of a non-supervised cluster classifier
	Boosting	This feature defines if the primary study is based or composed of an ensemble classifier with a boosting scheme
	Linear algorithm	This feature defines if the primary study is based on or composed of a classifier based on a linear algorithm
	Based on attributes	This feature defines whether the primary study is based or composed of a classifier based on attributes
	Multiclass	This feature defines if the primary study is based or composed of a multiclass classifier
	Warehouse	This feature defines if the primary study is composed of a classifier based on data warehouse
	SVM vectors	This feature defines if the primary study is composed of a classifier that makes use of support vector machines
	Neighbor method	This feature defines if the primary study is composed of a classifier based on the neighbor method
	Opinion-based	This feature defines if the primary study is composed of an opinion-based classifier
	Target-based	This feature defines if the primary study is composed of a target-based classifier
	Random forest	This feature defines if the primary study is composed of a random forest classifier
RQ4	Novelty	This feature defines if the primary study is a new proposal that does not exist in the literature
	Analysis	This feature defines if the primary study is an analysis of several existing proposals in the literature
	Research	This feature defines if the primary study is an investigation of existing or new proposals
	Improvement	This feature defines if the primary study is an improvement of an existing proposal in the literature

Table 6: Industrial classification scheme.

Research Question	Feature	Description
RQ1	Tool	This feature defines if the technology found is a tool for supporting
	Library	This feature defines if the technology found is a library of methods or framework
	System	This feature defines if the technology found is a complete system
	Platform	This feature defines if the technology found is a platform
	API	This feature defines if the technology found is an API that offers its functionalities
RQ2	Free	This feature defines whether the technology found is free software
	Commercial	This feature defines if the technology found is proprietary software
RQ3	*Python*	This feature defines whether the technology found is based on python
	Apache Spark	This feature defines whether the technology found is based on Apache Spark
	Node	This feature defines whether the technology found is based on node
	Java	This feature defines whether the technology found is based on java
	Ruby	This feature defines whether the technology found is based on ruby
	Unknown	This feature defines if the technology found does not allow knowing in what language it is based
RQ4	Recommendation	This feature defines if the technology found is aimed at the recommendation
	Classification	This feature defines if the technology found is aimed at the classification

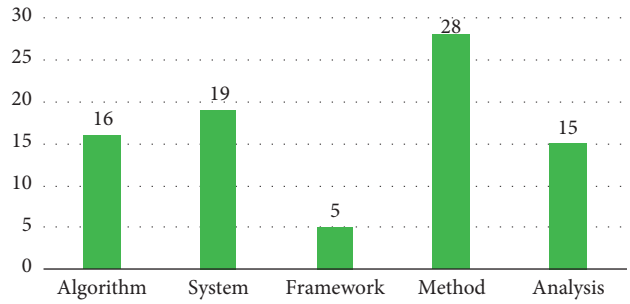


Figure 4: Scientific field: Research Question 1.

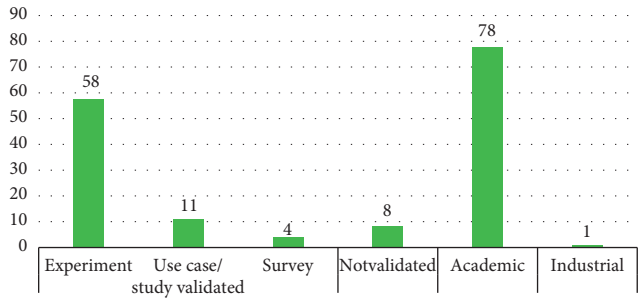


Figure 5: Scientific field: Research Question 2.

(i) Figure 8 shows the trend of publication in topics related to the classification and recommendation systems. The chart shows that the trend increases in recent years, so it can be deduced that it is a subject of high interest to the scientific community. It is important to note that, at the beginning of 2019, there are already more than half of the papers selected for the previous year.

(ii) Figure 9 presents the number of papers obtained for each of the digital libraries and the relationship with those finally selected for further study. In light green, the initial results are shown, highlighting ACM with 27 papers shown, followed by SCOPUS and IEEE Xplore with 23 and 14, respectively. ScienceDirect returned only 4 results. Dark green shows the finally selected studies of each digital library.

8.2. Industrial Report. After the description of the results obtained from the scientific report, this section presents the report of the data bring about conducting the study of the industrial scope.

(i) Research Question RQ1 finds the products that have been developed for the classification and recommendation systems. Figure 10 shows that the most frequent results have been complete systems and

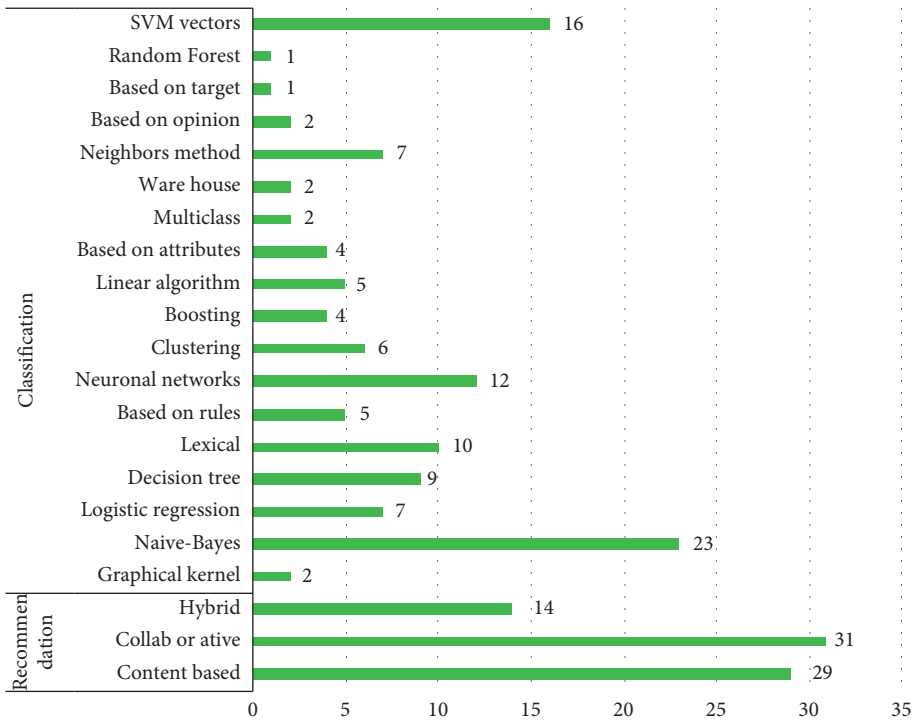


Figure 6: Scientific field: Research Question 3.

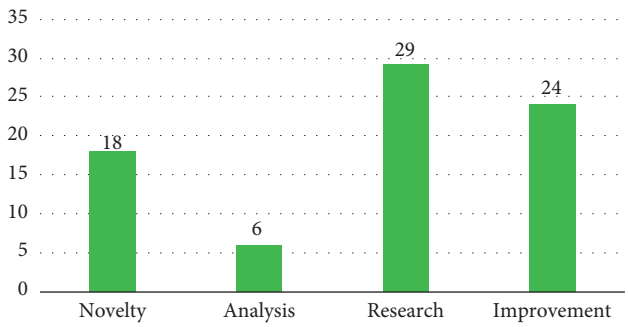


Figure 7: Scientific field: Research Question 4.

libraries or frameworks, with 5 and 4 proposals, respectively. The next two features are the APIs and tools, representing 3 and 4 proposals, respectively. In the last place, it located the platform feature, with just one proposal found.

The sum of the complete systems and the libraries represent 47.62% of the total of the proposals. The set of technologies that represent the APIs is 14.29%, the tools 9.52%, and finally, the platform is 4.76% of the total. From a software development life-cycle perspective (and avoiding methodological discussion), requirements and analysis phases differ from the design phase because it is an earlier stages and closer to the business (or application model) and is completely technology independent. Then, the found works are contextualized in the technological design phase. No contextualized work was found in early stages (business requirements or analysis).

Research Question RQ2 aims to determine if the products obtained in this scope are free or proprietary software. This classification has great interest to know those that can suppose an extra cost for the execution of the project.

According to the taxonomy defined, Figure 11 shows that results are balanced to the open side; commercial software, with 8 proposals, represent 38.10% of the total, and the set of free software technologies is composed of 12 results, 57.14% of the total.

(ii) Research Question RQ3 seeks to identify the nature of the products found. According to the taxonomy carried out after the extraction of features, results obtained are shown in Figure 12. It has been found that there is a group that gathers most of the technologies. This group corresponds to *Python*, with 7 results, representing 33.33% of the total. The next group with the highest results is *R*, with 28.57% after returning 6 results. After that, Java is placed, representing the 19.05% of total. Next, Apache Spark technology is classified with 3 proposals obtained, 14.29% of the total. Finally, there are two technologies with a single appearance, and they are Node and Ruby, with 9.52% of the total proposals found.

Within this research question, it is highlighted that a large amount of proprietary software did not allow to know what technology they are based on so they were included in the category of others. This category turned out to be 14.29% of the results, with 3 proposals.

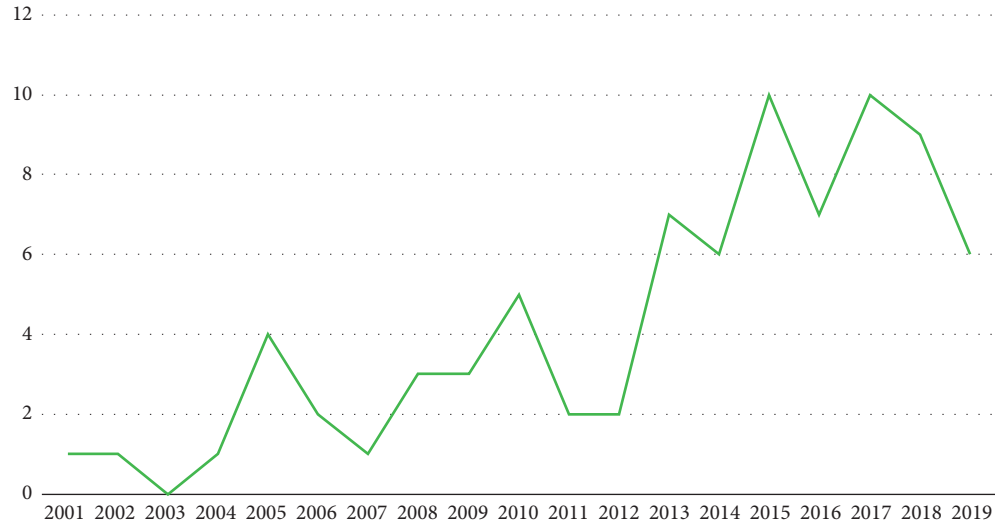


Figure 8: Results per year.

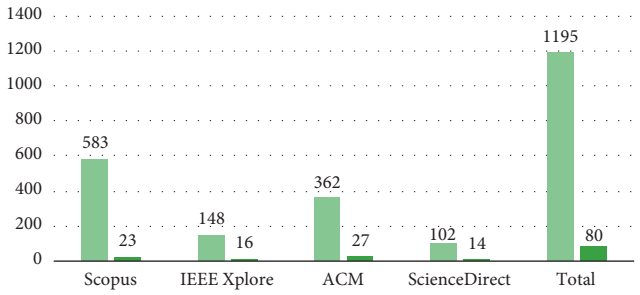


Figure 9: Selected by database.

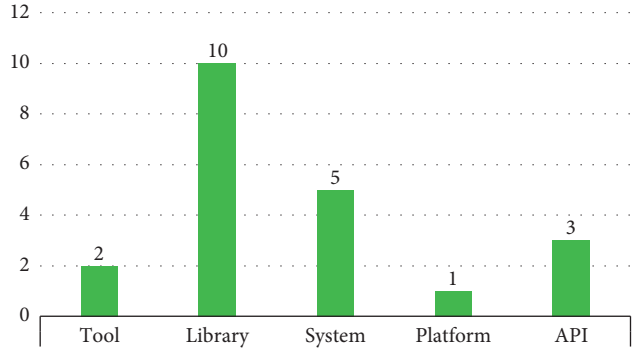


Figure 10: Industrial scope: Research Question 1.

(iii) Research Question RQ4 locates the main objective of the technology. In this case, two different groups have been stablished: classification and recommendation systems (Figure 13). In the case of the technologies that offer a classification system, a total of 10 proposals was obtained, representing 47.62% of the technologies implemented. In the case of recommendation systems, 76.19% of the technologies offered a solution to this problem; that is, 16 of the proposals were found. Finally, it is important to note that the 28.57% (6 proposals) of the total use both regression and classification.

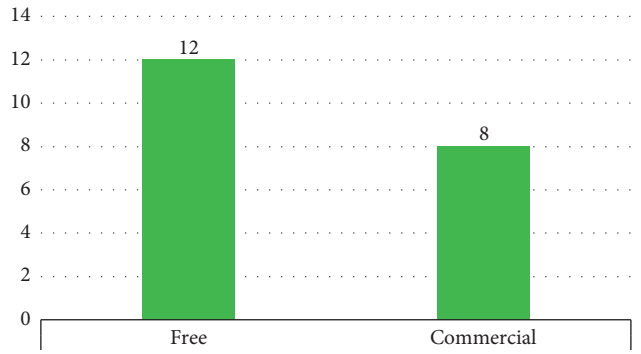


Figure 11: Industrial scope: Research Question 2.

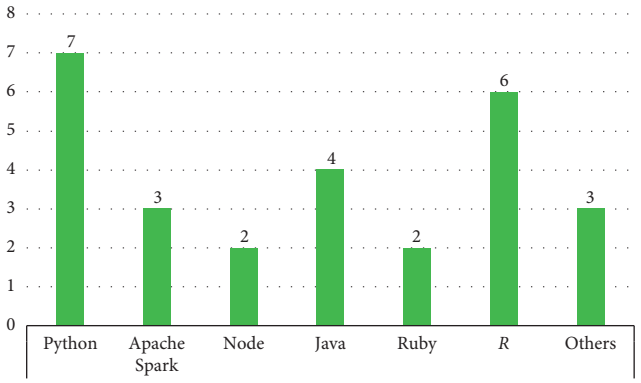


Figure 12: Industrial scope: Research Question 3.

9. Conclusion and Future Work

The development of this research has meant an immersion in the depths of the recommendation and classification systems, presenting a SMS which aims to illustrate the state of the art of these systems nowadays. In addition, with the execution of this study, it has been intended to offer help in decision-making about the algorithms to be implemented in the ADAGIO project.

Figure 13: Industrial scope: Research Question 4.

Unlike most SMS, that are focused on the scientific literature, and this study has been carried out from two points of view as discussed throughout the paper: the scientific and the industrial scopes.

A total of 80 primary studies obtained from the main digital libraries were analyzed. Within the scientific field, the results showed that the most studied technique in recommendation systems is recommendation with the use of collaborative filters, closely followed by those that use content-based filters. Only 14 used hybrid recommendation systems, whereas 31 used collaborative filtering and 29 used content-based methods. This is an interesting suggestion for researchers starting to use recommender systems, to find which of them are more popular and more used in the scientific environment. As there are more recommender systems than classification models, it seems that recommendation is well known for scientific researchers, and the most used technique is collaborative filtering.

In the case of classification solutions, the most researched alternatives correspond to naive Bayes, SVM vectors, and neuronal networks, representing almost 55% of the techniques used for this purpose. These results are due to the great presence of studies oriented to social networks, which cover a large part of Internet traffic.

It is important to point out that all the studies analyzed in the scientific field were found to be of a theoretical nature; i.e., none of them are within the industrial scope. Although many of the proposals present a validation, few of them use real data sources instead of synthetic ones (artificially generated rather than generated by real-world events) to carry out their experiments. In this sense, a lack of technology transfer of these proposals to real case studies has been detected.

Furthermore, by conducting market research through systematic industrial mapping, it was found that there are many technologies that offer automatic learning solutions, and most of which are complete systems or libraries. However, the nature of most of them could not be known because the proprietary software did not allow it. Another important issue that must be highlighted is that not only the communities of free software developers are interested in this topic but also there are large companies that are working on it for commercial purposes. This clearly shows the underlying economic interest, an indicator that it is a branch of long-distance research.

During the execution of the research on this subject, few studies were discovered that offered improvements to specific problems through the combination of recommendation and classification systems, the main motivation for this work. In the literature analyzed, the most interesting solutions, algorithms, and technologies have been found also to be used independently for classification and regression. This research is not only useful for the researcher trying to use both models at the same time but also for the analysts trying to do just classification or just regression. As future work, a very interesting research line may focus on how to combine these systems to obtain more efficient and effective solutions.

From a software development life-cycle perspective (and avoiding methodological discussions), requirements and analysis phases differ from the design phase because it is an earlier stages and closer to the business (or application model) and is completely technology independent. This SMS shows that the majority of all work carried out in the ML research and industrial field (combining classification and recommendation algorithms) respond to the design and implementation phase but are far from offering solutions in earlier stages such as requirements and analysis. This makes it very difficult to find efficient and effective solutions that support real business needs from an early stage. The present work let justify the opening of new ML research lines to support the information system development since early stages. A hypothetical solution proposal could be to provide business analysts with theoretical frameworks and support tools that facilitate the efficient and effective resolution of problems and that, subsequently, will allow the automation of their design and implementation. Specifically, this solution could consist of the definition of a theoretical framework:

9.1. Foundational Knowledge

(i) *Archetype Models for the Different Application Domains*. This model is used for the conceptualization, formalization, and categorization of the application domains under study. The objective is to understand which application domains exist and which is the basic information structure that should support the application domain. Through the development of these predefined archetype models, information structures could be offered in a systematic way in order to offer support to the different existing problems.

(ii) *Classification and Recommendation Template Methods to be Applied to Archetype Models*. This model is used for the conceptualization, formalization, and categorization of ML solutions (combining classification and recommendation algorithms) for all those application domains that have been defined by means of archetype models. The objective is to facilitate the development of a framework that allows the automatic generation of ML solutions and that, in addition, could adjust the classification and the recommendation according to the needs of each application domain.

9.2. Applied Knowledge

(i) From a strategic point of view, understanding the strategy as a set of ordered stages or phases (phase 1: classification and phase 2: recommendation)

Define ML solution strategies based on the combination of classification algorithms and recommendation. In other words, determine to what extent and in what manner (iterative and iterative-incremental) the classification and recommendation phases should be combined for a more efficient and effective use of these algorithms in problem solving. In addition, the above strategies may depend on the application domain being studied.
Determine which strategic configurations are most appropriate for each application domain. The idea is to facilitate decision-making by automating decisions by entering a particular application domain or problem.

(ii) From a tactical point of view

Determine which machine learning methods, techniques, and tools are the most effective and efficient for the application of the previous strategies, determining the most appropriate for each phase (classification and recommendation) according to the application domain of the object of study.

Finally, we can accomplish that even having executed this rigorous study, there is still a big difficulty in deciding about which algorithm is better than another depending on the context in which it is used. There is no generic classifier or recommender, and several should be implanted depending on the type of data. Currently, it also depends on the desired level of complexity and the cost of misclassification. In conclusion, there is no better model, and everything depends on the characteristics of each problem. In this sense, another possible future work is to characterize these systems, with formal methods (e.g., QuEF [111]), to reduce the cost when making decisions about it.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research has been supported by the Pololas project (TIN2016-76956-C3-2-R) of the Spanish Ministry of Economy and Competitiveness, the ADAGIO (P106-16/E09) project of the Centro para el Desarrollo Tecnológico Industrial (CDTI) of Spain, the Agencia Estatal de Investigación, Spain (Project MTM2017-86875-C3-2-R), and Gobierno de Extremadura, Spain (Project GR18108).

References

[1] S. Jaysri, J. Priyadharshini, P. Subathra, and Dr. (Col.) P. N. Kumar, "Analysis and performance of collaborative filtering and classification algorithms," *International Journal of Applied Engineering Research*, vol. 10, pp. 24529–24540, 2015.

[2] M. D. Ekstrand, J. T. Riedl, and J. A. Konstan, "Collaborative Filtering Recommender Systems," *Foundations and Trends® in Human—Computer Interaction*, vol. 4, no. 2, pp. 81–173, 2011.

[3] A. Gunawardana and G. Shani, "A survey of accuracy evaluation metrics of recommendation tasks," *Journal of Machine Learning Research*, pp. 2935–2962, 2009.

[4] M. Poussevin, V. Guigue, and P. Gallinari, "Extracting a vocabulary of surprise by collaborative filtering mixture and analysis of feelings," in *Proceedings of the CORIA 2015—Conference in Search Infomations and Applications—12th French Information Retrieval Conference*, Paris, France, March 2015.

[5] M. Z. Kurdi, "Lexical and syntactic features selection for an adaptive reading recommendation system based on text complexity," in *Proceedings of the 2017 International Conference on Information System and Data Mining*, pp. 66–69, Charleston, SC, USA, April 2017.

[6] M. A. Ghazanfar and A. Prügel-Bennett, "An improved switching hybrid recommender system using naive Bayes classifier and collaborative filtering," in *Proceedings of the International MultiConference of Engineers and Computer Scientists 2010 (IMECS)*, Hong Kong, China, 2010.

[7] A. I. Schein, A. Popescul, L. H. Ungar, and D. M. Pennock, "Methods and metrics for cold-start recommendations," in *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval—SIGIR '02*, New York, NY, USA, 2002.

[8] M. Ghazanfar and A. Prügel-Bennett, "Building switching hybrid recommender system using machine learning classifiers and collaborative filtering," *IAENG International Journal of Computer Science*, vol. 37, no. 3, 2010.

[9] Z. Hailong, G. Wenyan, and J. Bo, "Machine learning and lexicon based methods for sentiment classification: a survey," in *Proceedings of the 11th Web Information System and Application Conference (WISA)*, pp. 262–265, Tianjin, China, September 2014.

[10] R. Mu, "A survey of recommender systems based on deep learning," *IEEE Access*, vol. 6, pp. 69009–69022, 2018.

[11] I. Portugal, P. Alencar, and D. Cowan, "The use of machine learning algorithms in recommender systems: a systematic review," *Expert Systems with Applications*, vol. 97, pp. 205–227, 2018.

[12] B. Ouhbi, B. Frikh, E. Zemmouri, and A. Abbad, "Deep learning based recommender systems," *IEEE International Colloquium on Information Science and Technology (CiSt)*, vol. 2018, pp. 161–166, 2018.

[13] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system: a survey and new perspectives," *ACM Computing Surveys*, vol. 52, no. 1, p. 5, 2019.

[14] B. Kitchenham and S. Charters, "Guidelines for performing systematic literature reviews in software engineering," *Engineering*, vol. 2, p. 1051, 2007.

[15] K. Petersen, R. Feldt, S. Mujtaba, and M. Mattsson, "Systematic mapping studies in software engineering," in *Proceedings of the 12th International Conference on Evaluation and Assessment in Software Engineering*, vol. 17, p. 10, Bari, Italy, 2008.

[16] L. Leydesdorff, "Top-down decomposition of the journal citation report of the social science citation index: graph- and factor-analytical approaches," *Scientometrics*, vol. 60, no. 2, pp. 159–180, 2004.

[17] J. L. C. Izquierdo, V. Cosentino, and J. Cabot, "Analysis of co-authorship graphs of CORE-ranked software conferences," *Scientometrics*, vol. 109, no. 3, pp. 1665–1693, 2016.
[18] SCIE, "La Sociedad Científica Informática de España," 2017.
[19] SCIE, "GII-GRIN-SCIE (GGS) Conference Rating,"2019.
[20] A. Sattar, M. A. Ghazanfar, and M. Iqbal, "Building accurate and practical recommender system algorithms using machine learning classifier and collaborative filtering," *Arabian Journal for Science and Engineering*, vol. 42, no. 8, pp. 3229–3247, 2017.
[21] T.-D. Nguyen, T.-D. Cao, and L.-G. Nguyen, "DGA botnet detection using collaborative filtering and density-based clustering," in *Proceedings of the Sixth International Symposium on Information and Communication Technology*, pp. 203–209, Hue City, Vietnam, December 2015.
[22] T. Xie, Y. Chen, L. Hu, C. Gao, C. Hu, and J. Shen, "A multistage collaborative filtering method for fall detection," in *Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN)*, Rio, Brazil, August 2017.
[23] N. Thilagavathi and R. Taarika, "Content based filtering in online social network using inference algorithm," in *Proceedings of the 2014 International Conference on Circuits, Power and Computing Technologies (ICCPCT)*, Nagercoil, India, March 2014.
[24] X. Su, T. M. Khoshgoftaar, X. Zhu, and R. Greiner, "Imputation-boosted collaborative filtering using machine learning classifiers," in *Proceedings of the 2008 ACM Symposium on Applied Computing—SAC '08*, Fortaleza, Ceará, Brazil, March 2008.
[25] T. Shrot, A. Rosenfeld, J. Golbeck, and S. Kraus, "CRISP -an interruption management algorithm based on collaborative filtering," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Toronto, Canada, 2014.
[26] X. Zheng, "A credit scoring model based on collaborative filtering," in *Proceedings of the 9th International Conference on Computational Intelligence and Security*, Emei Mountain, Sichuan, China, December 2013.
[27] J. Li, H. Xu, X. He, J. Deng, and X. Sun, "Tweet modeling with LSTM recurrent neural networks for hashtag recommendation," in *Proceedings of the International Joint Conference on Neural Networks*, Vancouver, British Columbia, Canada, 2016.
[28] P. Liu, J. Cao, X. Liang, and W. Li, "A two-stage cross-domain recommendation for cold start problem in cyber-physical systems," in *Proceedings of the International Conference on Machine Learning and Cybernetics*, Guangzhou, China, 2015.
[29] P. Bedi, Richa, S. K. Agarwal, and V. Bhasin, "ELM based imputation-boosted proactive recommender systems," in *Proceedings of the 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Jaipur, India, September 2016.
[30] R. H. Nidhi and B. Annappa, "Twitter-user recommender system using tweets: a content-based approach," in *Proceedings of the ICCIDS 2017 International Conference on Computational Intelligence in Data Science*, pp. 1–6, Chennai, India, June 2017.
[31] R. Mittal and V. Sinha, "A personalized time-bound activity recommendation system," in *Proceedings of the 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC)*, Las Vegas, USA, January 2017.
[32] A. S. Vairagade and R. A. Fadnavis, "Automated content based short text classification for filtering undesired posts on Facebook," in *Proceedings of the IEEE World Conference on Futuristic Trends in Research and Innovation for Social Welfare (WCTFTR)*, Coimbatore, India, 2016.
[33] W. Bhebe and O. P. Kogeda, "Shilling attack detection in collaborative recommender systems using a meta learning strategy," in *Proceedings of the 2015 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC)*, pp. 56–61, IEEE, Windhoek, Namibia, May 2015.
[34] L. Bhatia and S. S. Prasad, "Building a distributed generic recommender using scalable data mining library," in *Proceedings of the 2015 IEEE International Conference on Computational Intelligence and Communication Technology (CICT)*, Ghaziabad, India, 2015.
[35] C. Biancalana, F. Gasparetti, A. Micarelli, A. Miola, and G. Sansonetti, "Context-aware movie recommendation based on signal processing and machine learning," in *Proceedings of the 2nd Challenge on Context-Aware Movie Recommendation*, Chicago, IL, USA, 2011.
[36] T. Zhang and V. S. Iyengar, "Recommender systems using linear classifiers," *Journal of Machine Learning Research*, pp. 313–334, 2002.
[37] V. Pronk, W. Verhaegh, A. Proidl, and M. Tiemann, "Incorporating user control into recommender systems based on naive Bayesian classification," in *Proceedings of the ACM International Conference on Recommender Systems*, Minneapolis, MN, USA, 2007.
[38] R. Burke, B. Mobasher, C. Williams, and R. Bhaumik, "Classification features for attack detection in collaborative recommender systems," in *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining—KDD '06*, Philadelphia, PA, USA, August 2006.
[39] Y. Song, L. Zhang, and C. L. Giles, "Automatic tag recommendation algorithms for social recommender systems," *ACM Transactions on the Web*, vol. 5, no. 1, p. 31, 2011.
[40] Y. M. Brovman, "Optimizing similar item recommendations in a semi-structured marketplace to maximize conversion," in *Proceedings of the 10th ACM Conference on Recommender Systems—RecSys '16*, Boston, MA, USA, September 2016.
[41] S. E. Middleton, D. C. De Roure, and N. R. Shadbolt, "Capturing knowledge of user preferences," in *Proceedings of the International Conference on Knowledge capture—K-CAP*, Victoria, BC, Canada, 2001.
[42] P. P. Jean-Jacques, J. Noack, and K. Bodarwé, "Emotion-based music recommendation using supervised learning," in *Proceedings of the 14th International Conference on Mobile and Ubiquitous Multimedia*, Linz, Austria, December 2015.
[43] A. Thor and E. Rahm, "AWESOME—A Data Warehouse-Based System for Adaptive Website Recommendations," in *Proceedings of the Thirtieth International Conference on Very Large Data Bases*, vol. 30, pp. 384–395, VLDB Endowment, Toronto, Ontario, Canada, September 2004.
[44] Y. H. Gu, S. J. Yoo, Z. Piao, J. No, Z. Jiang, and H. Yin, "A smart-device news recommendation technology based on the user click behavior," in *Proceedings of the Sixth International Conference on Emerging Databases: Technologies, Applications, and Theory*, pp. 9–16, Jeju Island, Republic of Korea, October 2016.
[45] X. Li and H. Chen, "Recommendation as link prediction in bipartite graphs: a graph kernel-based machine learning approach," *Decision Support Systems*, vol. 54, no. 2, pp. 880–890, 2013.
[46] A. A. Kothari and W. D. Patel, "A novel approach towards context based recommendations using support vector

machine methodology," *Procedia Computer Science*, vol. 57, pp. 1171–1178, 2015.

[47] W. P. Lee, C. T. Chen, J. Y. Huang, and J. Y. Liang, "A smartphone-based activity-aware system for music streaming recommendation," *Knowledge-Based Systems*, vol. 131, pp. 70–82, 2017.

[48] D. Han, J. Li, W. Li, R. Liu, and H. Chen, *An App Usage Recommender System: Improving Prediction Accuracy for Both Warm and Cold Start Users*, Multimedia Systems, 2019.

[49] A. Visuri, R. Poguntke, and E. Kuosmanen, *Proposing Design Recommendations for an Intelligent Recommender System Logging Stress*, Association for Computing Machinery, New York, NY, USA, 2018.

[50] E. R. Núñez-Valdez, D. Quintana, R. G. Crespo, P. Isasi, and E. Herrera-Viedma, "A recommender system based on implicit feedback for selective dissemination of ebooks," *Information Sciences*, vol. 467, pp. 87–98, 2018.

[51] S. Narayan and E. Sathiyamoorthy, "A novel recommender system based on FFT with machine learning for predicting and identifying heart diseases," *Neural Computing and Applications*, vol. 31, no. S1, pp. 93–102, 2019.

[52] A. Pujahari and V. Padmanabhan, "An approach to content based recommender systems using decision list based classification with k-DNF rule set," in *Proceedings of the 2014 13th International Conference on Information Technology (ICIT)*, Bhubaneswar, India, December 2014.

[53] M. Mehdi, N. Bouguila, and J. Bentahar, "Probabilistic approach for QoS-aware recommender system for trustworthy web service selection," *Applied Intelligence*, vol. 41, no. 2, pp. 503–524, 2014.

[54] R. A. Gotardo, E. R. Hruschka, S. D. Zorzo, and P. R. M. Cereda, "Approach to cold-start problem in recommender systems in the context of web-based education," in *Proceedings of the 2013 12th International Conference on Machine Learning and Applications (ICMLA)*, Miami, FL, USA, December 2013.

[55] H. Costa, B. Furtado, D. Pires, L. Macedo, and A. Cardoso, "Context and intention-awareness in POIs recommender systems," in *Proceedings of the 6th ACM Recommender Systems Conference, 4th Workshop on Context-Aware Recommender Systems (RecSys)*, vol. 12, p. 5, Dubai, UAE, September 2012.

[56] U. Rohini and V. Ambati, "A collaborative filtering based reranking strategy for search in digital libraries," in *Lecture Notes in Computer Science*, Springer, Berlin, Germany, 2005.

[57] Y. Z. Wei, L. Moreau, and N. R. Jennings, "Learning users' interests by quality classification in market-based recommender systems," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 12, pp. 1678–1688, 2005.

[58] W. Paireekreng, "Mobile content recommendation system for re-visiting user using content-based filtering and client-side user profile," in *Proceedings—International Conference on Machine Learning and Cybernetics*, Lanzhou, China, 2013.

[59] S. Lu, B. Wang, H. Wang, and Q. Hong, "A hybrid collaborative filtering algorithm based on KNN and gradient boosting," in *Proceedings of the 13th International Conference on Computer Science and Education (ICCSE)*, Colombo, Sri Lanka, August 2018.

[60] L. Zhang, B. Xiao, J. Guo, and C. Zhu, "A scalable collaborative filtering algorithm based on localized preference," in *Proceedings of the 7th International Conference on Machine Learning and Cybernetics (ICMLC)*, Melbourne, Australia, December 2008.

[61] S. Feng, M. Zhang, Y. Zhang, and Z. Deng, "Recommended or not recommended? Review classification through opinion extraction," in *Proceedings of the 12th Asia-Pacific Web Conference, Advances in Web Technologies and Applications (APWeb)*, Busan, Korea, April 2010.

[62] B. Alghofaily and C. Ding, "Meta-feature based data mining service selection and recommendation using machine learning models," in *Proceedings of the 2018 IEEE 15th International Conference on e-Business Engineering (ICEBE)*, Xi'an, China, October 2018.

[63] C. Yang, S. Ren, Y. Liu, H. Cao, Q. Yuan, and G. Han, "Personalized channel recommendation deep learning from a switch sequence," *IEEE Access*, vol. 6, pp. 50824–50838, 2018.

[64] M. Tkalčič, A. Odić, A. KoTkalšičir, and J. Tasič, "Affective labeling in a content-based recommender system for images," *IEEE Transactions on Multimedia*, vol. 15, no. 2, pp. 391–400, 2013.

[65] A. A. Kothari and W. D. Patel, "A novel approach towards context sensitive recommendations based on machine learning methodology," in *Proceedings of the 2015 5th International Conference on Communication Systems and Network Technologies (CSNT)*, Gwalior, MP, India, April 2015.

[66] R. Trepos, A. Salleb, M. O. Cordier, V. Masson, and C. Gascuel, "A distance-based approach for action recommendation," in *Lecture Notes in Computer Science*, Springer, Berlin, Germany, 2005.

[67] J. S. Pedro and S. Siersdorfer, "Ranking and Classifying Attractiveness of Photos in Folksonomies," in *Proceedings of the 18th International Conference on World Wide Web*, pp. 771–780, ACM, Madrid, Spain, April 2009.

[68] T. Raeder, T. R. Hoens, and N. V. Chawla, "Consequences of variability in classifier performance estimates," in *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, Sydney, Australia, 2010.

[69] J. J. Ahn, S. J. Lee, K. J. Oh, T. Y. Kim, H. Y. Lee, and M. S. Kim, "Machine learning algorithm selection for forecasting behavior of global institutional investors," in *Proceedings of the 42nd Annual Hawaii International Conference on System Sciences (HICSS)*, Waikoloa, Hawaii, January 2009.

[70] D. Arendt, E. Saldanha, R. Wesslen, S. Volkova, and W. Dou, "Towards rapid interactive machine learning: evaluating tradeoffs of classification without representation," in *Proceedings of the 24th International Conference on Intelligent User Interfaces*, pp. 591–602, Marina del Ray, CA, USA, March 2019.

[71] A. G. C. de Sá and G. L. Pappa, "Towards a method for automatically evolving bayesian network classifiers," in *Proceedings of the 15th Annual Conference Companion on Genetic and Evolutionary Computation*, pp. 1505–1512, ACM, Amsterdam, Netherlands, July 2013.

[72] K. Zhao and L. Pan, "A machine learning based trust evaluation framework for online social networks," in *Proceedings of the 2014 IEEE 13th International Conference on Trust, Security and Privacy in Computing and Communications*, Beijing, China, September 2014.

[73] E. Dufourq and B. A. Bassett, "Automated problem identification: regression vs. classification via evolutionary deep networks," in *Proceedings of the South African Institute of Computer Scientists and Information Technologists*, p. 12, Thaba Nchu, South Africa, September 2017.

[74] B. F. De Souza, A. C. P. L. F. De Carvalho, and C. Soares, "Empirical evaluation of ranking prediction methods for gene expression data classification," in *Lecture Notes in Computer Science*, Springer, Berlin, Germany, 2010.
[75] M. Unger, B. Shapira, L. Rokach, and A. Bar, "Inferring contextual preferences using deep auto-encoding," in *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization*, pp. 221–229, ACM, Bratislava, Slovakia, July 2017.
[76] W. Yunli, "Automatic recognition of text difficulty from consumers health information," in *Proceedings of the IEEE Symposium on Computer-Based Medical Systems*, Salt Lake City, Utah, 2006.
[77] R. Vainshtein, A. Greenstein-Messica, G. Katz, B. Shapira, and L. Rokach, "A hybrid approach for automatic model recommendation," in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pp. 1623–1626, ACM, Turin, Italy, October 2018.
[78] L. Jiang and H. Zhang, "Learning instance greedily cloning naive Bayes for ranking," in *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, p. 8, IEEE, Houston, TX, USA, 2005.
[79] Z. Qiao, S. Zhao, C. Xiao, X. Li, Y. Qin, and F. Wang, "Pairwise-ranking based collaborative recurrent neural networks for clinical event prediction," in *Proceedings of the IJCAI International Joint Conference on Artificial Intelligence*, Stockholm, Sweden, July 2018.
[80] R. Ali, S. Lee, and T. C. Chung, "Accurate multi-criteria decision making methodology for recommending machine learning algorithm," *Expert Systems with Applications*, vol. 71, pp. 257–278, 2017.
[81] R. Lafta, J. Zhang, X. Tao et al., "A general extensible learning approach for multi-disease recommendations in a telehealth environment," *Pattern Recognition Letters*, 2018.
[82] S. Bag, S. K. Kumar, and M. K. Tiwari, "An efficient recommendation generation using relevant jaccard similarity," *Information Sciences*, vol. 483, pp. 53–64, 2019.
[83] A. Soudani and W. Barhoumi, "An image-based segmentation recommender using crowdsourcing and transfer learning for skin lesion extraction," *Expert Systems with Applications*, vol. 118, pp. 400–410, 2019.
[84] S. S. Durduran, "Automatic classification of high resolution land cover using a new data weighting procedure: the combination of *k*-means clustering algorithm and central tendency measures (KMC-CTM)," *Applied Soft Computing*, vol. 35, pp. 136–150, 2015.
[85] C. L. Chi, W. N. Street, and M. M. Ward, "Building a hospital referral expert system with a prediction and optimization-based decision support system algorithm," *Journal of Biomedical Informatics*, vol. 41, no. 2, pp. 371–386, 2008.
[86] N. Pombo, N. Garcia, and K. Bousson, "Classification techniques on computerized systems to predict and/or to detect apnea: a systematic review," *Computer Methods and Programs in Biomedicine*, vol. 140, pp. 265–274, 2017.
[87] J. Szymański and J. Rzeniewicz, "Identification of category associations using a multilabel classifier," *Expert Systems with Applications*, vol. 61, pp. 327–342, 2016.
[88] J. Pinho Lucas, S. Segrera, and M. N. Moreno, "Making use of associative classifiers in order to alleviate typical drawbacks in recommender systems," *Expert Systems with Applications*, vol. 39, no. 1, pp. 1273–1283, 2012.
[89] R. Espinosa, D. García-Saiz, M. Zorrilla, J. J. Zubcoff, and J. N. Mazón, "S3mining: a model-driven engineering approach for supporting novice data miners in selecting suitable classifiers," *Computer Standards & Interfaces*, vol. 65, pp. 143–158, 2019.
[90] D. Cournapeau, "Scikit-learn," 2019.
[91] N. Hug, "Surprise," 2019.
[92] M. Kula, "LightFM," in *Proceedings of the 2nd Workshop on New Trends on Content-Based Recommender Systems Co-Located with 9th ACM*, Vienna, Austria, September 2015.
[93] K. Vand, "Rexy," 2019.
[94] A. S. Foundation, *PredictionIO*, A. S. Foundation, Pune, Maharashtra, 2019.
[95] G. Jenson, "HapiGER," 2019.
[96] L. C. A. and Credits, "LensKit," 2019.
[97] I. SuggestGrid, "SuggestGrid," 2019.
[98] S. Systems, "SLI Systems Recommender," 2019.
[99] A. W. Services, "AmazonWebService Machine Learning," 2019.
[100] Microsoft, "Azure ML Studio," 2019.
[101] Gravity Research & Development, "Yusp," 2019.
[102] IBM Watson Studio, "IBM Watson," 2019.
[103] Recombee, "Recombee," 2019.
[104] Mr. Dlib, "Mr. DLib," 2019.
[105] Caret, "Caret," 2019.
[106] Shiny, "Shiny," 2019.
[107] RandomForest, "RandomForest," 2019.
[108] KlaR, "KlaR," 2019.
[109] CORElearn, "CORElearn," 2019.
[110] RecommenderLab, "RecommenderLab," 2019.
[111] F. J. Domínguez-Mayo, M. J. Escalona, and M. Mejías, "QuEF (quality evaluation framework) for model-driven web methodologies," in *Lecture Notes in Computer Science*, Springer, Berlin, Germany, 2010.

Hindawi
Scientific Programming
Volume 2019, Article ID 8319549, 16 pages
https://doi.org/10.1155/2019/8319549

Research Article

A High-Frequency Data-Driven Machine Learning Approach for Demand Forecasting in Smart Cities

Juan Carlos Preciado,[1] **Álvaro E. Prieto**,[1] **Rafael Benitez**,[2] **Roberto Rodríguez-Echeverría**,[1] **and José María Conejero**[1]

[1]*Dept. Ingeniería Sistemas Informáticos y Telemáticos, Universidad de Extremadura, Cáceres, Extremadura, Spain*
[2]*Dept. Matemáticas para la Economía y la Empresa, Universidad de Valencia, Valencia, Spain*

Correspondence should be addressed to Álvaro E. Prieto; aeprieto@unex.es

Received 7 March 2019; Revised 3 May 2019; Accepted 13 May 2019; Published 3 June 2019

Academic Editor: Can Özturan

Different types of sensors along the distribution pipelines are continuously measuring different parameters in Smart WAter Networks (SWAN). The huge amount of data generated contain measurements such as flow or pressure. Applying suitable algorithms to these data can warn about the possibility of leakage within the distribution network as soon as the data are gathered. Currently, the algorithms that deal with this problem are the result of numerous short-term water demand forecasting (WDF) approaches. However, in general, these WDF approaches share two shortcomings. The first one is that they provide low-frequency predictions. That is, most of them only provide predictions with 1-hour time steps, and only a few provide predictions with 15 min time steps. The second one is that most of them require estimating the annual seasonality or taking into account not only data about water demand but also about other factors, such as weather data, that make their use more complicated. To overcome these weaknesses, this work presents an approach to forecast the water demand based on pattern recognition and pattern-similarity techniques. The approach has a twofold contribution. Firstly, the predictions are provided with 1 min time steps within a time lead of 24 hours. Secondly, the laborious estimation of annual seasonality or the addition of other factors, such as weather data, is not needed. The paper also presents the promising results obtained after applying the approach for water demand forecasting to a real project for the detection and location of water leakages.

1. Introduction

The current big data scenario is based on using a large volume of data to get new insights and acquire knowledge that support the daily decision-making process [1]. One of the main sources of these data are IoT (Internet of Things) systems that collect and transfer a great amount of sensor data [2]. The use of these technologies for water management allows gathering data in order to monitor water usage and water waste, what is regarded as one of the application areas of a smart city [3]. In this sense, the application of information and communication technology (ICT) devices to water distribution systems (WDSs) is considered a key subarea of a smart city and introduces the concept of Smart WAter Network (SWAN) [4]. A SWAN consists of a large number of sensors that measure automatically and continously a wide range of parameters present in WDS.

It should be noted that WDSs are big and complex. Only in Europe, there are more than 3.5 million kilometers of pipes [5], and in the United States, around 159 billion liters of water are withdrawn from water sources each day [6]. The management of WDS implies to deal with different issues. One of them is the problem of water pressure that could affect significantly the level of service for the users and where there are novel approaches such as [7] that proposes the division of the network in subregions according to the expected water peak demand.

Another huge problem managing WDS is to deal with water loss. Water loss can be attributed to several causes, including leakage, metering errors, and fraud although

leakage is usually the major cause. It is estimated that the amount of water in the world that is lost is more than 30 percent of production [8].

The data obtained by the sensors that compose a SWAN can be an important turning point to avoid this problem. This is due to the fact that the usual gathered data include flow, pressure, or totalizer measurements. The application of water demand forecasting algorithms over all these data allows detecting leakages at an early stage.

There are several works that present different approaches to try to forecast the water demand applying different techniques. Due to the necessity to detect a water leakage as soon as this problem arises, the more suitable approaches are those with a short-term forecast horizon, that is, how far the prediction about the future demand is able to accurately reach. Thus, a short-term forecast horizon is generally considered for a range between 1 and 48 hours.

The existing short-term water demand forecast approaches can achieve good results. However, in general, they have in common two important limitations that the approach proposed in this work reduces.

The first limitation refers to its frequency, in other words, how many predictions within this horizon the approach is able to provide. The usual time steps of most of the approaches are 1 hour, so that a frequency of 24 predictions per day may be achieved. Only a few approaches provide higher frequency being, at most, one prediction every 15 minutes. Considering that the sooner the prediction is able to detect an anomaly, the better any improvement in the frequency of the predictions could significantly reduce the loss of water. Although the time horizon of our approach is on average (24 hours), we are able to get a time step of one minute, that is, a frequency of 1440 each day, without reducing the accuracy of the prediction. Notice that this is not a trivial contribution because we identified that neural networks approaches were unfeasible with this frequency and more classic methods such as ARIMA and dynamic harmonic regression were even too computationally expensive.

The second limitation concerns the data needed apart from previous water demand. Most of the current approaches need extra data about weather (temperature, rainfall, etc) or demand changes according to factors related to weekly or annual seasonality, being particularly the estimation of the latter, annual seasonality, a very demanding task. Our approach uses previous water demand data just considering weekly seasonality reduction and thus the complexity of its application. Therefore, it avoids the troublesome estimation and inclusion of annual seasonality or the usage of weather data.

Our approach is based on pattern similarity and is inspired by the work of Grzegorz Dudek [9–11] for short-term load forecasting in the daily operation of power systems and energy markets. It has been implemented using the model-driven development (MDD) paradigm [12, 13] and has been tested in one of the partner cities of the European project SmartWater4Europe [5]. The following goodness-of-fit (GoF) parameters have been used to determine the performance of the approach: MAPE (mean average percentage error), RMSE (root mean squared error), and FOB (fraction out of bounds).

It should also be emphasized that this approach not only reduces both aforementioned limitations but also presents the next advantage: (a) it is relatively easy to implement; (b) it is not highly time-consuming; (c) as the historical record increases, the performance improves; and (d) the method is robust enough to deal with minor data issues such as small segments of missing data. The latter avoids that it causes "false alarms".

The rest of the paper is organized as follows. In Section 2, we review previous work on water demand forecasting. Section 3 describes the locations where the data were gathered and the proposed algorithm. In Section 4, we present the results and discussion. Finally, the conclusions and future work are outlined in Section 5.

2. Related Work

Water demand has been a field where quantitative forecasting has been applied profusely because it meets the twofold requirement [14] to use this kind of forecasting: (a) there are historical numerical data about the variable to forecast and (b) it is plausible to presuppose that some features of the patterns recognized in the historical data are recurring.

We found a number of water demand forecasting approaches proposed in the literature. In this sense, there are works published during the 1990s that can be considered as fundamentals in this field such as the ones by Shvartser et al. [15] or Buchberger et al. [16, 17]. Donkor et al. [18] reviewed the literature on urban water demand forecasting published from 2000 to 2010, in order to identify the methods and models that are useful for specific water utility decision-making problems. More recently, Sebri [19] conducted a meta-analysis to estimate in a statistical way how different features of primary studies could influence the correctness of urban water demand forecasts.

In this section, we focus on reviewing the most relevant methods published since 2010 to date (to the best of our knowledge) focused on short-term predictions (1–48 hours) sorted according to the frequency used (from lowest to highest).

To begin with, Adamowski et al. [20] tested if coupled wavelet-neural network models (WA-ANNs) applied to forecast daily urban water demand could provide promising results during the summer months in the city of Montreal, Canada. They used daily total urban water demand, daily total precipitation, and daily maximum temperature, all of them gathered during the summer period to conduct their work. Concretely, they integrated artificial neural networks together with discrete wavelet transforms to elaborate coupled wavelet-neural network models. They stated that their approach provided better results forecasting short-term (24 hours) water demand than other techniques such as artificial neural networks (ANN) alone, autoregressive integrated moving average (ARIMA), multiple linear regression (MLR), or multiple nonlinear regression (MNLR).

However, their approach only provided one prediction for the whole day.

Herrera et al. [21] focused their work on trying to forecast the water demand in the next hour in an urban area of a city in southeastern Spain. Not only did they use previous water demand data but also temperature, wind velocity, atmospheric pressure, and rain data. They concluded that support vector regression (SVR) models were the more adequate ones for this task, and multivariate adaptive regression splines (MARS), projection pursuit regression (PPR), and random forest (RF) could also be used. However, the neural network that they used (feedforward neural networks with one hidden layer in conjunction with the backpropagation learning algorithm) seemed to provide very poor results.

Odan and Reis [22] compared different ANNs to forecast water demand. They used hourly consumption data from the water supply system of Araraquara, São Paulo, Brazil, as well as temperature and relative humidity data. Their estimations were made for the next 24 hours with a frequency of 1 for each hour. Concretely, they analyzed a multilayer perceptron with the backpropagation algorithm (MLP-BP), a dynamic neural network (DAN2), and two hybrid ANNs. The more interesting finding of their work is that the different variants of DAN2 that they used either to forecast the first hour or the whole 24 hours did not need the use of weather variables and achieved better results that the rest ones.

Ji et al. [23] used different factors along with a least-square support vector machine (SVM) to forecast water demand for one day with one-hour frequency. The factors that they have taken into account were flow data, the maximum and the minimum temperature, precipitations, holiday information, and information of incidents. The novelty of this work lies in the adjustment of the hyperparameters of the SVM system by using swarm intelligence via a teaching learning-based optimization algorithm.

Hutton and Kapelan [24] were concerned about the uncertainties that influenced the results of water demand forecasts and proposed an iterative methodology based on probabilistic that tried to decrease the effect of such uncertainties during the development of hourly short-term water demand prediction models. They used static calendar data in addition to water demand data. On the one hand, their approach exposed the unsuitability of simplistic Gaussian residual assumptions in predicting water demand. On the other hand, they concluded that a model whose kurtosis and heteroscedasticity in the residuals are revised iteratively using formal Bayesian likelihood functions allow building better predictive distributions.

Candelieri et al. [25–27] have works that make use of unsupervised (time series clustering) and supervised (support vector machines regression models) machine learning strategies. These strategies were combined in a two-stage framework in order to identify typical urban water demand patterns and successively provide reliable one day forecasts for each hour of the day. They used real data gathered from different sources of Milan (Italy) to check their proposal. Their last work extended the previous ones by allowing also anomaly detection.

Alvisi and Franchini [28] have the goal of estimating the predictive uncertainty in water demand forecasting. To this end, they joined short-term water demand predictions provided by two or more models by means of the model conditional processor (MCP). Then, MCP computed a probability distribution of the real future demand according to the different predictions of each particular model. This probability distribution, together with a predefined hourly pattern based on the season and the day of the week, allows them to estimate the expected hourly water demand for a whole day as well as the associated predictive uncertainty.

Brentan et al. [29] considered that the result of the applying fixed regression structure with time series can be biased and prone to errors. Their proposal tried to reduce both of them when building a short-term (24 hours) hourly water demand forecasting. To do this, firstly, they used support vector regression (SVR) together with calendar data to build a base forecasting, and secondly, they improved this forecasting applying Fourier time series process.

Romano and Kapelan [30] proposed the use of evolutionary artificial neural networks (EANNs) to perform adaptive hourly water demand forecasting for the whole next day. Their goal is to provide near real-time operational management by analyzing water demand time series and weekly seasonality. This approach was tested on a real-life UK case study, and one of its main features was that it did not need too much human intervention.

Gagliardi et al. [31] proposed two models based on homogeneous Markov chain model (HMC) and non-homogeneous Markov chain model (NHMC) to forecast next day hourly water demand. They used water demand data and weekly seasonality; concretely, they differentiated between working and nonworking days. They recommended the use of HMC to do this type of predictions because their results showed that its performance was better than the one obtained using NHMC.

Pacchin et al. [32] proposed a model based on moving windows that predicted the hourly water demand during the next day. This model presented two different features with respect to other similar models. On the one hand, it updated the prediction taking into account the demand data of the previous day. On the other hand, it did not need two much historic data in comparison with other models since it was able to do accurate predictions only using the data of three or four previous weeks. It also should be pointed that they also took into consideration the weekly seasonality.

Arandia et al. [33] proposed a methodology to predict 15 min, hourly, and daily water demand either offline (using historical data) or online (using a real-time feed of data). Their proposal joined seasonal ARIMA (SARIMA) and data assimilation. They also used in their approach weekly seasonality and daily periodicity and concluded than their methodology showed a better performance using weekly seasonality.

Bakker et al. [34] presented a model to forecast 15 min water demand for the next two days. Their model used static calendar data in addition to six years of water demand data gathered from different areas of the Netherlands. According to this work, a frequency of 15 minutes is more suitable than 1-hour frequency when detailed optimization is needed.

As we have seen, a number of approaches have been widely used for forecasting; however, as it is shown in Table 1, the frequency of these approaches is usually around 1 for each hour. Additionally, this table also shows the factors that each proposal needs to work apart from the previous water demand measurements. In most cases, the inclusion of more factors to make the forecast, such as annual calendar data or weather data, can be quite cumbersome. In turn, we propose the application of pattern similarity-based techniques proposed by Dudek [9–11] to the water demand forecasting problem. The main reason for selecting these techniques is their ability to simultaneously cope with the aforementioned difficulties: they remove the need to add weather data or to determine the annual seasonality by constructing the input and output patterns in which the series has been normalized, and at the same time, since the considered signal segments encompass a full day, the frequency of the predictions is 1440 per day.

3. Materials and Methods

This section describes the data sites used (taken from diverse real-world locations with different characteristics) and the preprocessing procedure carried out before starting the data analysis. In addition, we describe some relevant concepts, such as trends and seasonalities, before describing the input/output patterns and the proposed algorithm.

3.1. Data Sites. The algorithm has been tested in different locations of one of the member cities of the European project SmartWater4Europe. Concretely, this city is located in northern Spain, and it has about 180.000 inhabitants, with a population density of 1680 inhabitants per square km. With respect to the climate, it has an average annual precipitation of 546 mm, and it has a range of daily mean temperatures from 3.5°C in winter to 19.5°C in summer.

The data were collected by the company responsible of managing the water distribution of this city. This company has 58507 customers, the length of the distribution network is 467.315 metres, and the mean quantity of supplied water each day per inhabitant is 392,39 litres.

To gather the data, the company used the following:

(i) 14 sectoral sensors spread throughout the network pipes of the 3 sites (see below) that were able to measure flow, pressure, and totalizer each minute. This means that each sensor measures 1440 times a day and the company had been storing 20160 measurements (14×1440) each day for 10 years.

(ii) 1502 intelligent water meters spread throughout industries and homes located in the 3 sites (see below). In this case, each one performed 24 measurements per week.

Concretely, this company measured data of three different areas of this city whose characteristics are as follows.

(i) Site 1: Industrial Area. It is an industrial estate at the outskirts of the city. In this area, there are almost no domestic end users of the water supply.

(ii) Site 2: High-Density Population Area. It is a neighborhood located in the center of the city. It is a zone with high buildings where there are thousands of families.

(iii) Site 3: Low-Density Population Area. It is a suburb of the city. Most homes are either low-rise buildings or single-family homes, so the density of users is very low. It is important to note that the houses of this area have private backyards. It may be assumed that this is a factor which influences the water use pattern of the area.

At each timestamp, the minimum, maximum, and average flows (measured in l/min) were recorded. Table 2 shows, as an example, the first six measurements obtained by a sectoral sensor for the industrial area site. Note that the variable timestamp reflects the local time (CET), and +01 or +02 only reflects the difference from Greenwich Mean Time (GMT) (or coordinated universal time, abbreviated to UTC).

3.2. Proposed Algorithm. Domestic water demand data conform a time series with several seasonalities being the daily, weekly, and annual seasonalities the most important. In addition to these seasonalities, there are usually a long-term trend component and a high-frequency noise term.

As was mentioned before, the signal was sampled at a 1-minute frequency and we were considering a 24 hour forecast horizon. This means that, at any given moment, we need to forecast the next 1440 values of our signal. This rules out the possibility of using, directly, classical time series analysis methods such as ARIMA, exponential smoothing, and Winter–Holts methods. Moreover, direct neural network methods are also not feasible since for these methods the output layers would have 1440 neurons and the input layer would be much bigger, and therefore, the training of such large number of weights would require far more data than what is available.

The main problem here is that, with this high sampling frequency, the number of data needed in order to capture the weekly and annual seasonalities is simply too large. Therefore, we need to devise a method in which the seasonalities can be treated in a different way.

Our approach here is based on the pattern-similarity search proposed by Dudek in [9–11] for forecasting electric load. This method first splits the time series into segments of length equal to the forecast horizon and then maps those segments into two signals x and y—input and output signals—which will be used for a query-predict procedure. Those signals will be somehow normalized and will not be affected by trends and large period seasonalities. They will only contain the information within the forecast horizon (24 hours), and each 24-hour segment will be considered as a measure unit.

Table 1: Related work comparison with respect to frequency, forecast horizon, and other factors or complex estimation needed to apply the approach.

Work	Related work comparison		
	Frequency	Forecast horizon	Other factors
Adamowski et al. [20]	1 for each day	24 hours	Weather data during summer
Herrera et al. [21]	1 for each hour	1 hour	Weather data
Odan and Reis [22]	1 for each hour	24 hours	Weather data
Ji et al. [23]	1 for each hour	24 hours	Weather, holidays, and incident data
Hutton and Kapelan [24]	1 for each hour	24 hours	Annual calendar data
Candelieri et al. [25–27]	1 for each hour	24 hours	Working days and seasons of the year
Alvisi and Franchini [28]	1 for each hour	24 hours	Weekly seasonality and seasons of the year
Brentan et al. [29]	1 for each hour	24 hours	Annual calendar data
Romano and Kapelan [30]	1 for each hour	24 hours	Weekly seasonality
Gagliardi et al. [31]	1 for each hour	24 hours	Weekly seasonality
Pacchin et al. [32]	1 for each hour	24 hours	Weekly seasonality
Arandia et al. [33]	1 for each 15 minutes	24 hours	Daily and weekly seasonality
Bakker et al. [34]	1 for each 15 minutes	48 hours	Annual calendar data
Our proposal	1 for each minute	24 hours	Weekly seasonality

Table 2: Structure of the raw data from the industrial area (first six measurements).

	Industrial area		
Timestamp	Average	Maximum	Minimum
2014-01-01 00:00:00 + 01	3.278	4.031	2.919
2014-01-01 00:01:00 + 01	3.591	5.064	3.049
2014-01-01 00:02:00 + 01	4.875	5.352	4.518
2014-01-01 00:03:00 + 01	4.263	5.074	3.475
2014-01-01 00:04:00 + 01	3.966	5.004	3.406
2014-01-01 00:05:00 + 01	3.771	4.031	3.188
...	...	...	...

In particular, the input and output signals are defined by the following:

$$x_{i,t} = \frac{F_{i,t} - \overline{F}_i}{\sqrt{\sum_{l=1}^{n}\left(F_{i,l} - \overline{F}_i\right)^2}}, \quad t = 1, \ldots, n, \tag{1}$$

$$y_{i,t} = \frac{F_{i,t+\tau} - \overline{F}_i}{\sqrt{\sum_{l=1}^{n}\left(F_{i,l} - \overline{F}_i\right)^2}}, \quad t = 1, \ldots, n, \tag{2}$$

where $x_{i,t}$ and $y_{i,t}$ denote the input and output signals for day i at time t, respectively; $F_{i,t}$ denotes the water flow of the day i at time t; $\overline{F}_i$ denotes the average water flow of day i; τ is the forecast horizon; n is the number of measurements in each day (in our case, both τ and n are 1440).

On the one hand, the input signal x, also called the *query signal*, represents the normalized pattern for a current day with all its intraday information. On the other hand, the output signal y, also called the *forecast signal*, represents the normalized pattern of the following day (with our particular value of the forecast horizon). The normalization procedure filters all the seasonalities and trends beyond the daily frequency.

Now, the procedure would be as follows: for any given day i_0, we want to estimate the unknown value of the output signal y_{i_0} from the known input signal x_{i_0}. Once we have the estimation $\widehat{y}_{i_0}$, we can predict the values of the water demand for the forecast horizon using equation (2):

$$\widehat{F}_{i_0+1,t} = \widehat{F}_{i_0,t+\tau} = \overline{F}_{i_0} + \widehat{y}_{i_0,t}\sqrt{\sum_{l=1}^{n}\left(F_{i_0,l} - \overline{F}_{i_0}\right)^2}. \tag{3}$$

Therefore, the problem reduces to obtain the forecast for the output signal $\widehat{y}_{i_0}$. To make such forecast, we follow the next procedure (shown in Figure 1):

(1) We select the k nearest neighbors (using the Euclidean distance) of the query pattern x_{i_0} from the data in the history record from days of the same class (same day of the week/holiday) such that the following day is not an atypical day (e.g., holiday).

(2) We compute the estimate $\widehat{y}_{i_0}$ via the following equation:

$$\widehat{y}_{i_0} = \frac{1}{k}\sum_{j \in \Theta\left(x_{i_0}\right)} y_j, \tag{4}$$

where $\Theta(x_{i_0})$ is the set of indices of the k x patterns nearest to the query pattern x_{i_0} obtained in the previous step.

(3) Finally, we transform $\widehat{y}_{i_0}$ to obtain the water flow estimate according to equation (3).

Along with the estimation, we can obtain pointwise confidence bands:

$$I_{i_0,t} = \widehat{F}_{i_0+1,t} \pm T^*_{\alpha/2}\frac{S_{i_0,t}}{\sqrt{k}}\sqrt{\sum_{l=1}^{n}\left(F_{i_0,l} - \overline{F}_{i_0}\right)^2}, \quad t = 1, \ldots, n, \tag{5}$$

where $T^*_{\alpha/2}$ denotes the two-tailed critical value for Student's t-distribution with $k-1$ degrees for a confidence level α and $S^2_{i_0,t}$ is the sample variance of the k output signals used for the computation of $\widehat{y}_{i_0}$.

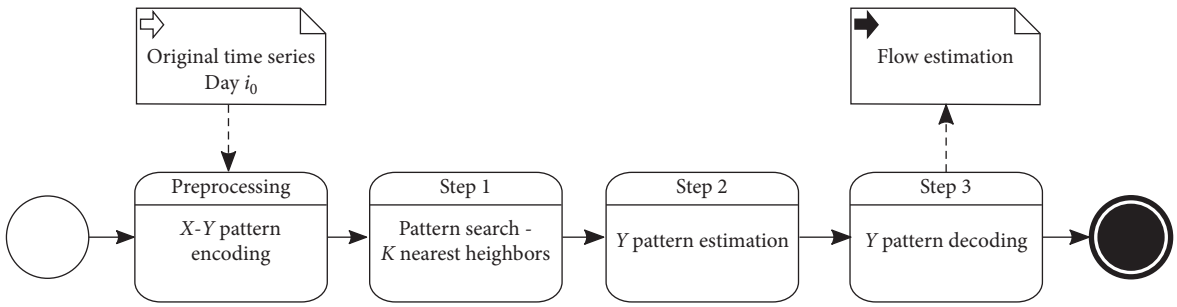


Figure 1: Forecasting procedure.

Once we have an estimation, we need to assess the quality of the forecast in order to validate the prediction model. We have considered three GoF parameters: the mean average percentage error (MAPE), the root mean squared error (RMSE), and the fraction out of bounds (FOB). The first two parameters are well-known error measures. The FOB for estimation at day i_0 is as follows:

$$\text{fob}_{i_0} = \frac{\text{NOB}_{i_0}}{\text{NIB}_{i_0} + \text{NOB}_{i_0}}, \quad (6)$$

where NOB_i and NIB_i are the number of measurements on day i_0 that lie outside and inside the confidence band for the given day i_0, respectively.

The MAPE parameter is very widely used in forecasting practice, but it becomes of little use when the actual values to be forecast are very small (close to zero). The problem is that, in our segments, there is a significant fraction of the day for which the water demand is indeed very small (night ours). The RMSE is an absolute value of the deviance of the forecast from the observed data. However, for small values of the water flow, it is difficult to assess the goodness of fit when the RMSE is small since the measure is not relative to the magnitude of the quantity to be predicted. Finally, the FOB can be regarded as a measure of the deviance of the observed day from what could be considered an average day of the same type. For small values of the FOB, we could say that the observed water flow corresponds to an "average" day, while if the FOB is large, the observed data does not follow the same pattern of other days of the same type in the historical record (and this could be related to either measurement anomalies or even water leaks).

4. Results and Discussion

4.1. Algorithm Parameters. Figure 2 depicts the average water flows vs. the day of the week for the three measurement sites. It is clearly shown that, for the industrial area site, there are at least four different patterns: Monday–Thursday, Friday, Saturday, and Sunday. For the high-density population area, there are two patterns corresponding to the labor days (Mon–Fri) and the weekends (Sat–Sun). Finally, at the low-density population area, there are more irregular patterns. Therefore, we considered the most restrictive pattern distribution (each day of the week to follow a different pattern, low-density population area) with the aim of easing the development of the algorithm.

Moreover, another distinct pattern is shown on holidays (Figure 3). Since the distribution of holidays varies from year to year, the inclusion of a holiday pattern in a model based on periodicities is difficult and cumbersome to implement.

The algorithm was tested for all days from 15 February 2014 until 18 September 2016. We did not start with the first measurements because we needed some weeks of historical data for the k nearest neighbors approach. Since we searched for the five nearest neighbors (see details below), we left a margin of seven weeks of historical data. The parameters considered for all sites were as follows: (a) the number of nearest neighbors: $k = 5$ for all years, (b) the threshold limits: min = 0.05 l/min, max = 100 l/min, and (c) the confidence level for the confidence band estimation: 90% (i.e., $\alpha = 0.1$).

The number of neighbors is an important parameter. If it is too small, the resulting pattern will not be representative of a true pattern for the forecasted day, but if it is too large, then the neighbors might be "far away" from the query pattern, and thus, we would be considering very different days for the estimation of our pattern.

4.2. Results. Figure 4 depicts a general perspective of all three GoF parameters at the three sites. All of them showed small global values: in 75% of the cases, the predictions showed values of FOB, MAPE, and RMSE less than 0.20, 39%, and 7.86 l/min, respectively, for the industrial area site; 0.21, 50%, and 1.80 l/min for the high-density population area site; and 0.25, 41%, and 5.82 l/min for the low-density population area site (Table 3).

Although the errors showed small values in the overall outcomes, a more thorough analysis is needed to determine the causes for the cases in which the parameters took higher values. To this aim, daily values were obtained and represented as scatter plots. For the sake of simplicity, values were categorized in three different levels: good, regular, and bad. The results are shown in Figure 5.

4.3. Discussion. The pattern-similarity forecasting method presented here proved very suitable for obtaining accurate daily predictions for the water flow values. However, we found several cases in which predictions were worse than what was expected or they simply delivered values completely different from the actual measurements.

At the industrial area site, the most difficult days to forecast were Sundays (for example, Figure 6). The main

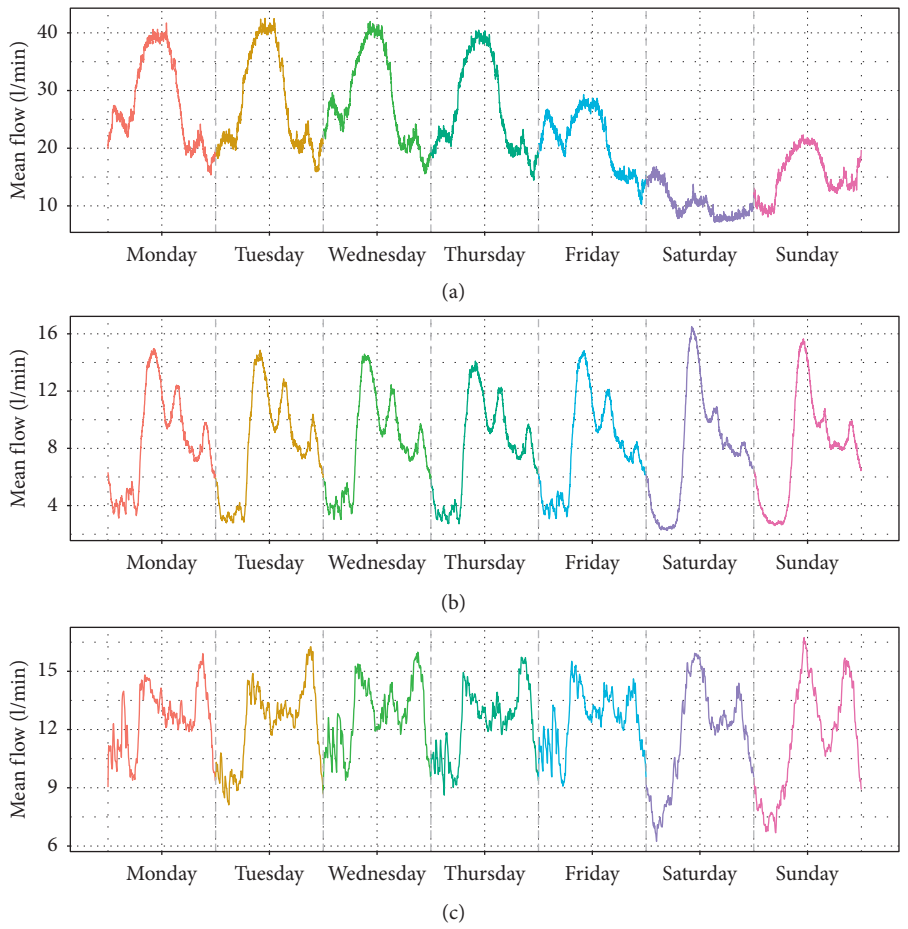


Figure 2: Mean flow vs. day of the week. For each of the three sites, the average for each minute of each weekday for the whole period is plotted. (a) Industrial area; (b) high-density population area; (c) low-density population area.

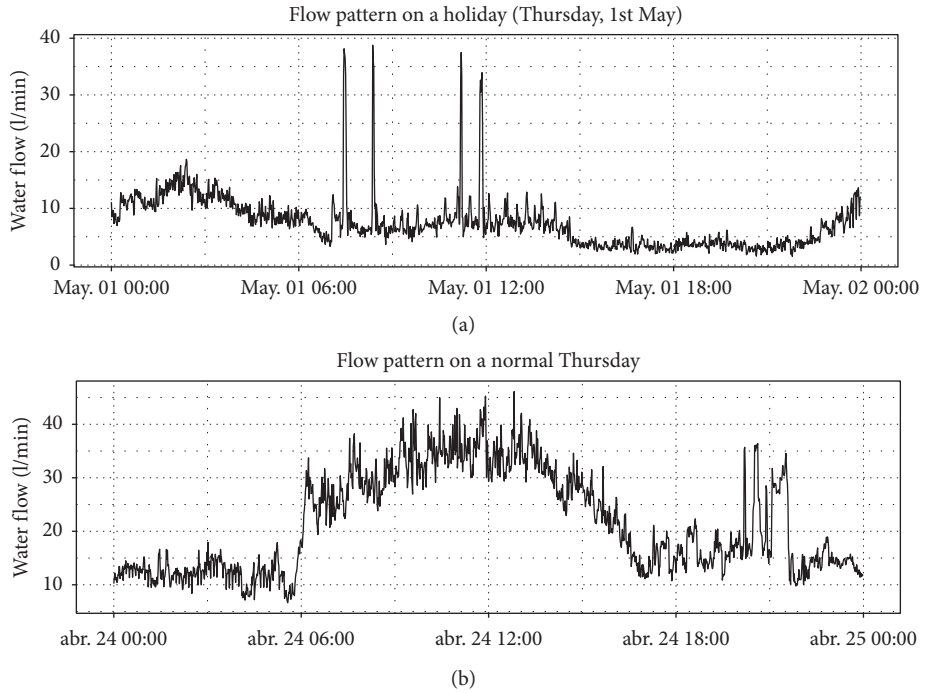


Figure 3: A typical holiday pattern (a) compared to a typical pattern for the same weekday on a nonholiday day (b).

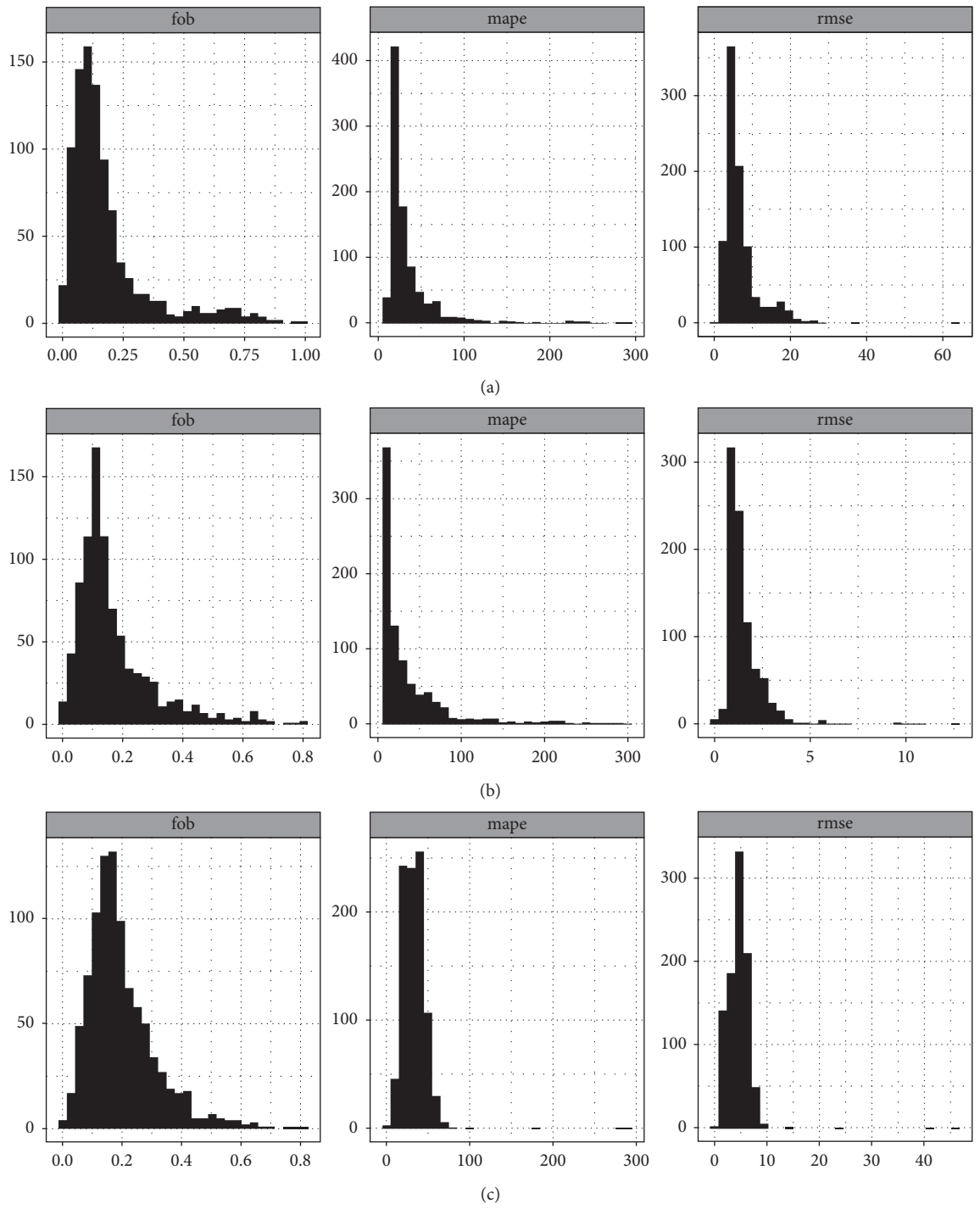


Figure 4: Distributions for the goodness of fit parameters values at the industrial area (a), the high-density population area (b), and the low-density population area (c) sites over the entire period of study, 2014–2016.

reason for this difficulty was that there were several different patterns for Sundays, but on the other hand, all Saturdays showed almost the same pattern. For any given Sunday, we took the preceding Saturday X pattern and we looked for the k nearest neighbors for this Saturday pattern. Since most of that patterns were very similar, regardless of the corresponding Y pattern (for next Sunday), there were cases in which the k Y patterns were almost random and did not reflect the characteristic Y pattern for our day. In other terms, a given X pattern had very different possible Y patterns linked to it.

Other difficulties arose when dealing with anomalous days. The most common anomalous days were holidays. We had an issue with the forecasting procedure, when the day we wished to forecast, the following day, or the preceding day was a holiday. Moreover, even if the prediction day was not a

Table 3: 75% quantiles for the goodness-of-fit parameters at the three sites over the entire period.

SITE	GOF		
	FOB (ratio)	MAPE (%)	RMSE (l/min)
Industrial area	0.20	39	7.86
High-density population area	0.21	50	1.80
Low-density population area	0.25	41	5.82

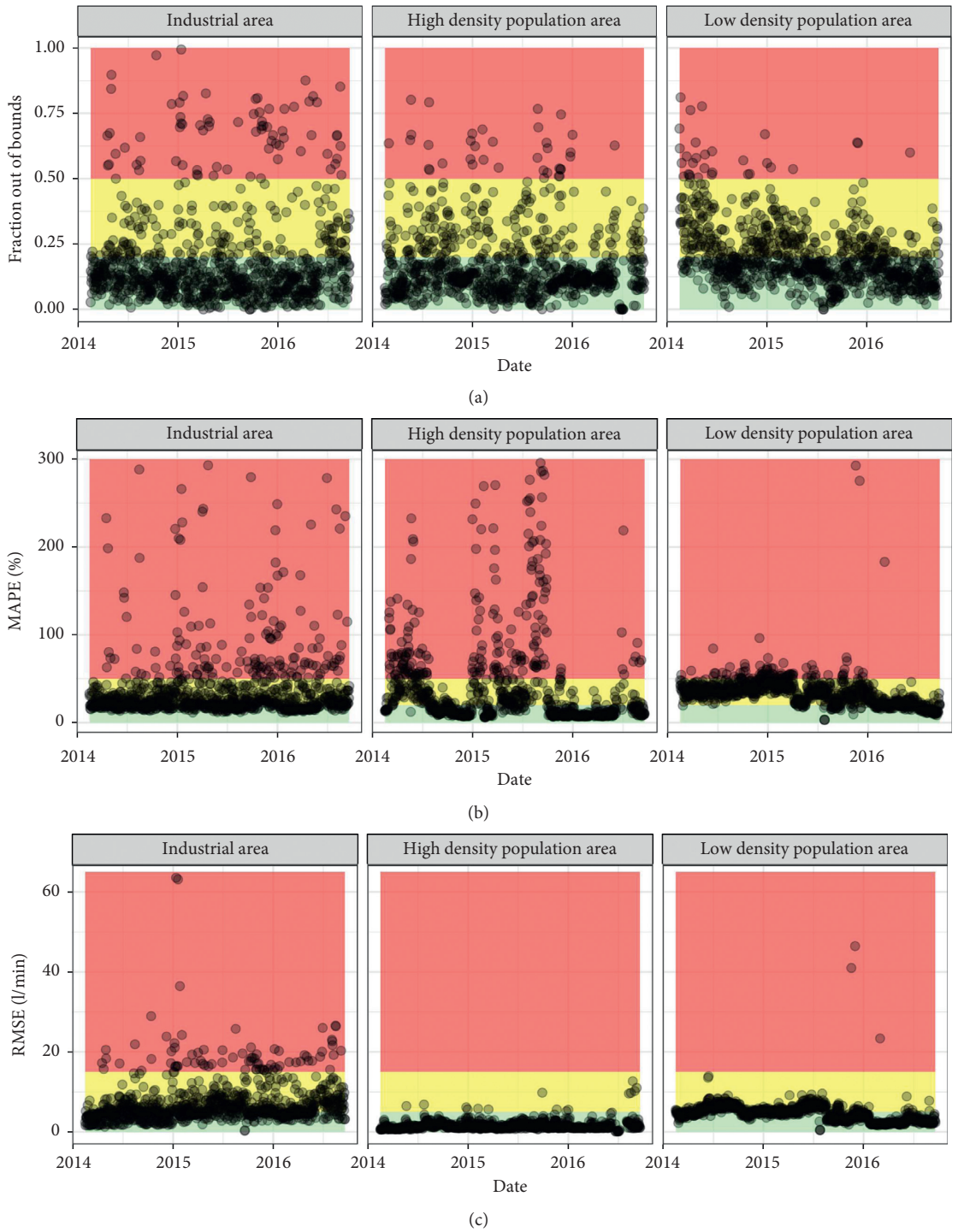


Figure 5: Daily values of the (a) FOB (fraction out of bounds), (b) MAPE (mean average percentage error), and (c) RMSE (root mean square error) for the industrial area, the high-density population area, and the low-density population area sites. The values are gathered into three different categories: good (green shading) for values lower than 20%, regular (green shading) for values between 20 and 50%, and bad (red shading) for values greater than 50%.

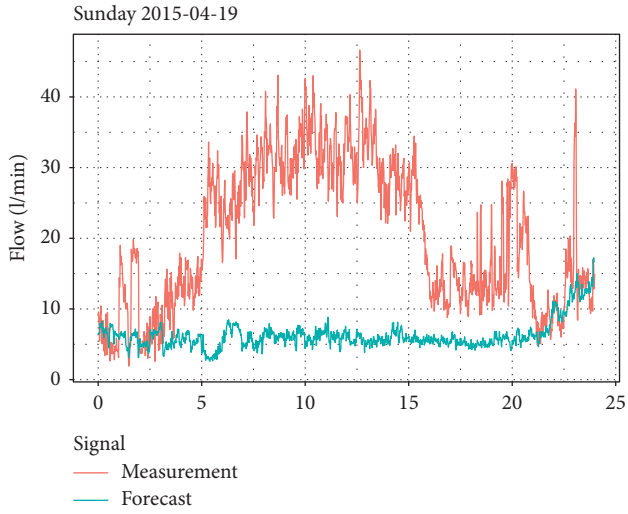


Figure 6: A wrongly forecasted signal. In this case, the forecasted day was on Sunday, 19 April 2015, at the industrial area site. For this site, Saturdays are very similar and Sundays differ depending on other factors (e.g., the time of the year). This led to an improper forecasting since the (k) Saturdays closest to 18 April 2015 do not need to be followed by Sundays similar to the forecasted day.

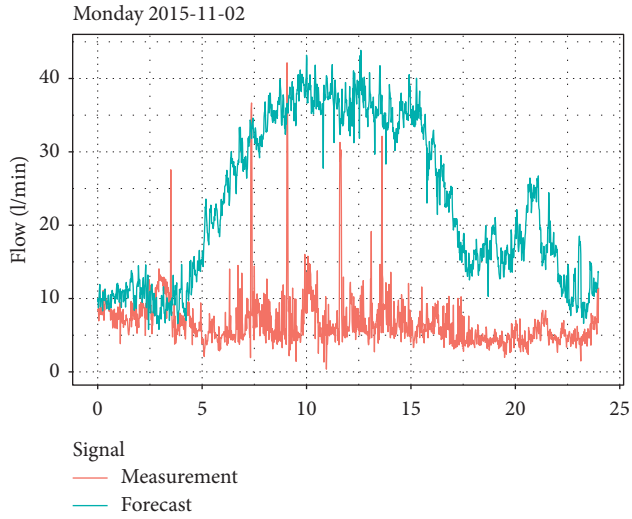


Figure 7: Forecasting at the industrial area on Monday, 2 November 2015. This day in particular was a holiday. In this case, since the day before was a Sunday, the predicted values were the average of (k) normal Mondays, which, obviously had a very different behaviour.

holiday (nor the preceding or following day), but one of the k nearest neighbors of the X pattern for the query day was followed by an anomalous day, the results were distorted by it (Figure 7).

Finally, the better the measurements are, the more accurate the forecasts will be. If a particular day has gaps in the measurements, although it is feasible to cope with the NAs (days with nonavailable data), they will introduce errors and mispredictions into the algorithm (Figure 8). Days with missing data should be flagged as anomalous and not considered in the forecasting procedure.

The high-density population area is the site where predictions were most accurate. This is partly because in this site which is an urban neighborhood in the center of the city, there are a large number of inhabitants living in residential buildings. This large number of people using the water supply at once regularizes the water flow time series. This is seen in Figure 2 (middle plot). Two patterns (labor days and weekends) can be observed, and they even look similar. Since the signal is so regular, the statistical forecasting procedure is more reliable, and thus, the GoF parameters showed very good results (with the exception of the MAPE which, as we stated above, was not considered due to its flaws when near-zero values are measured).

In the high-density population area, one of the causes of errors in the forecasting was misbehavior of the time series, probably because of malfunctions in the measurement device. For example, on 24 July 2014, there was, at around 10 a.m., a jump in the signal. After the jump, the time series continued to follow the normal pattern but maintained the offset (Figure 9).

Another type of misbehavior seen in the time series was an increase in the random fluctuations of the signal. For example, on 18 November 2015, a high-frequency random

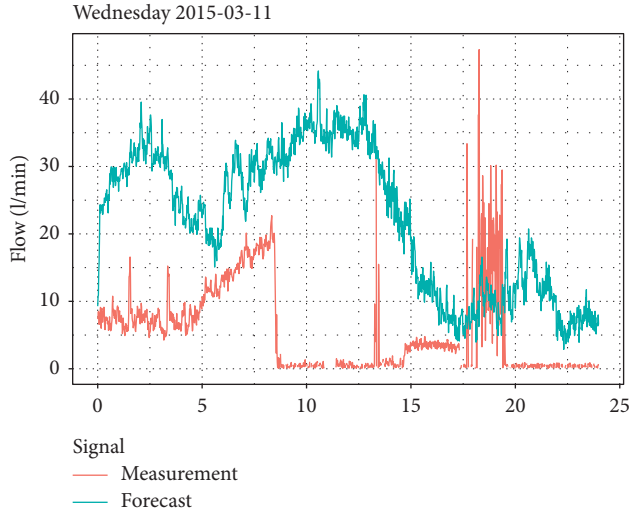


Figure 8: Forecasting at the industrial area on Wednesday, 11 March 2015. The effect of missing data in both the prediction and query days is visible in this figure. When there are missing data in the query day, the neighbors are improperly obtained since only the remaining data are considered, and thus, one might obtain a misleading neighbor which would yield erroneous forecasting results. When the data are missing in the forecasting day, there cannot accurate predictions.

component appeared and overlapped the signal (Figure 10). This random noise made the FOB value increase to 0.51 (red flag). Nonetheless, the original signal was still well predicted since both the forecasted and the measured values followed the same pattern, and this is why the RMSE value stayed small (RMSE = 2.8 l/min) although the FOB was high.

The low-density population area site was the most difficult to forecast. The reason for this was because this site is a

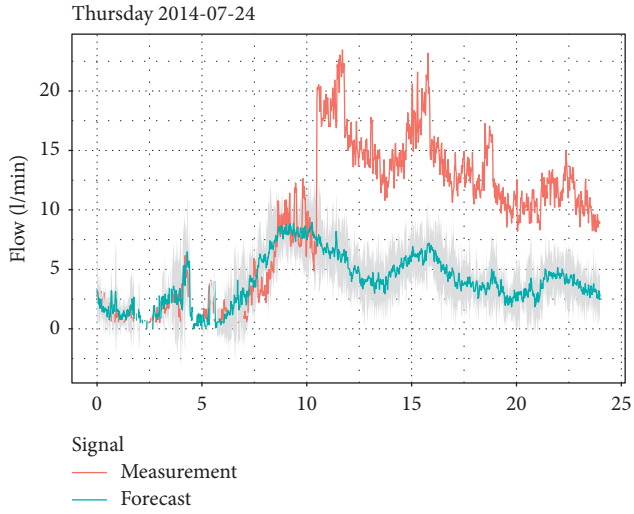


Figure 9: Forecasting at the high-density population area on Thursday, 24 July 2014. The water flow jumped at 10:28 a.m. from 9.39 l/min to 19.03 l/min. This offset of around 10 l/min was maintained after the jump. Therefore, although the RMSE was not very high (RMSE = 6.81 l/min), the FOB put that day in red (FOB = 0.63). The pointwise 95% confidence band of the estimation is depicted in grey.

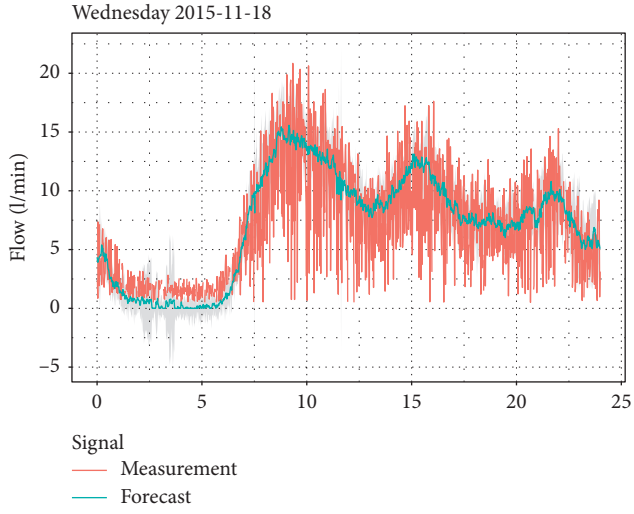


Figure 10: Forecasting at the high-density population area on Wednesday, 18 November 2015. The water flow measurements show a very large random component that increases the variance by a great amount. However, the general pattern of the series was well predicted by the forecast. Therefore, the RMSE remained low (RMSE = 2.8 l/min), even when the FOB was high (FOB = 0.51).

residential suburb of the city, where there are lots of single-family homes with gardens. Moreover, the zone is not very big, so the number of end users is very low compared to that of the high-density population area (hundreds of end users vs. thousands of end users). Therefore, the weight of each domestic user is very high and so is the variance in the signals. This led to a highly irregular time series. For example, Figure 11 shows the same week of July (8th to 14th) for 2014, 2015, and 2016. No regular pattern can be easily foreseen.

In this site, the overall results were fairly good. However, the main characteristic of these signals, the sudden peaks in the water flow, remained largely unpredictable since their distribution is, to a large extent, random.

For example, Figure 12 shows the forecast for two days. The first one corresponds to 12 March 2014 and was flagged as "green" in the FOP plot (Figure 5), while the second, which is for 13 March 2014, was flagged as "red" in the same FOP plot. It looks like the forecasts were more or less equally good, but in the first case, the peaks happened to occur inside the confidence bands, while in the second case, they fell outside those bands. This seems to be the reason why there were differences in the FOB for almost the same type of prediction even though the RMSE was, in both cases, very small (RMSE = 4.21 l/min, RMSE = 4.43 l/min).

As we have seen, an advantage of the pattern-similarity algorithm is that there is no need to estimate the annual seasonality since the procedure of normalizing the signal (obtaining the X–Y patterns) deseasonalizes the time series.

The pattern-similarity algorithm is easy to implement, and it is not very time-consuming. The part of the algorithm that takes the most time is the filtering of the training data, which are all days in the historical record of the same day of the week as the query day, such that neither them nor the next day are holidays (or anomalous days). The filtering can be accelerated by using database processing language tools (such as SQL).

In addition, if we have a large enough historical record with good data quality at our disposal, the patterns obtained as the forecast of the water flow on a given day are a good prediction of what that day should be like (Figure 13).

Even when the day to be forecasted shows minor data issues (small segments of missing data), the method is robust enough to deal with them. This keeps the algorithm from making false alarms (Figure 14).

Finally, as the historical record increases, the performance of the algorithm will improve. The method is based on obtaining statistical knowledge from previous data in order to determine the most similar situation to the one ahead, from the past data. Therefore, as the historical database gets larger, the forecast will get more accurate.

4.4. Comparison with Previous Work. In this section, we compare our approach (STPS, short-term pattern similarity) with another two similar ones: $\alpha\beta$-WDF ($\alpha\beta$ water demand forecast that was recently published by Pacchin et al. [32]) and GRNN (generalized regression neural network that was published by Dudek [11]).

The $\alpha\beta$-WDF approach is based on a moving window in which average parameters are obtained for similar days (same day of the week) from one, two, and three weeks earlier (a moving window of 3 weeks). $\alpha\beta$-WDF and STPS work in a very similar way; they both obtain average patterns for the same day of the week as the query day. However, since $\alpha\beta$-WDF takes into account three weeks prior to the time at which the forecast is made, STPS selects the k nearest

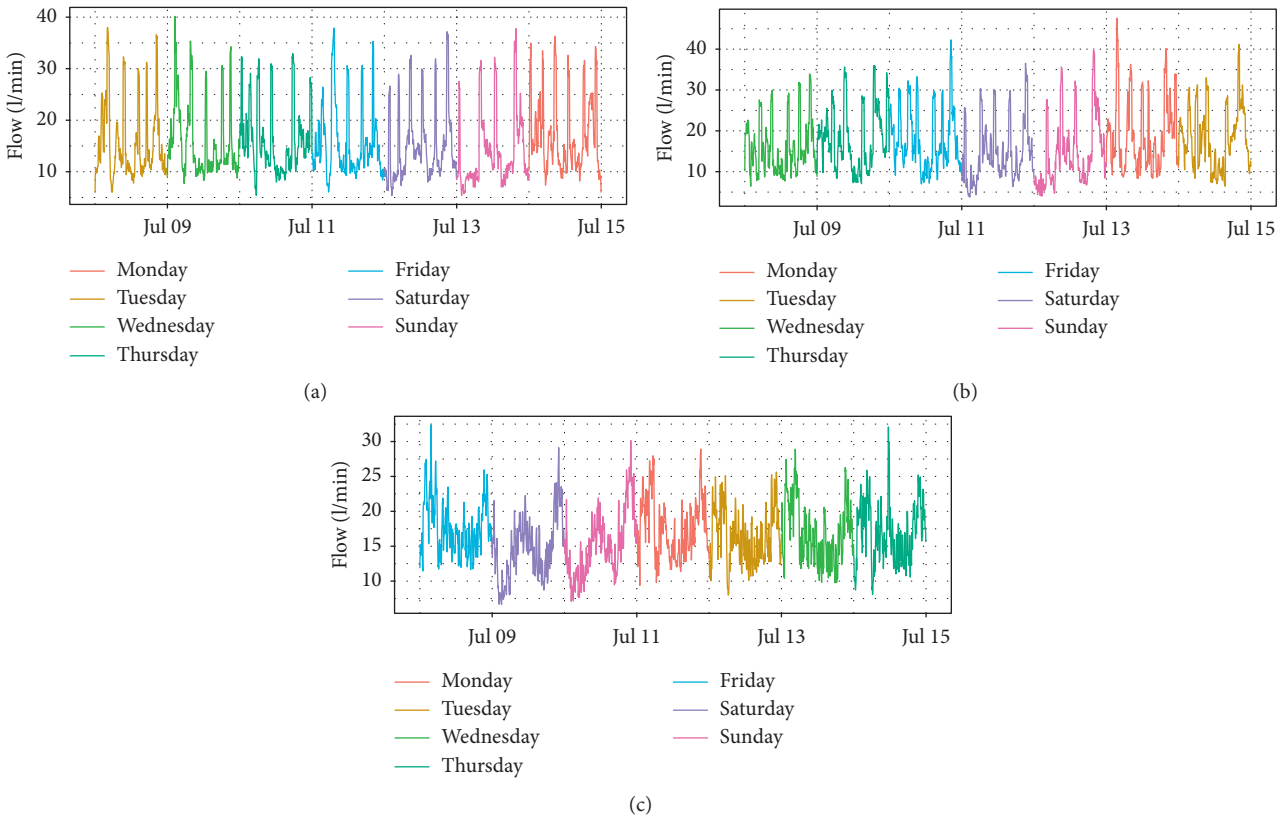


Figure 11: Comparison of the same week of July over the three sampled years (a) 2014, (b) 2015, and (c) 2016 at the low-density population area site.

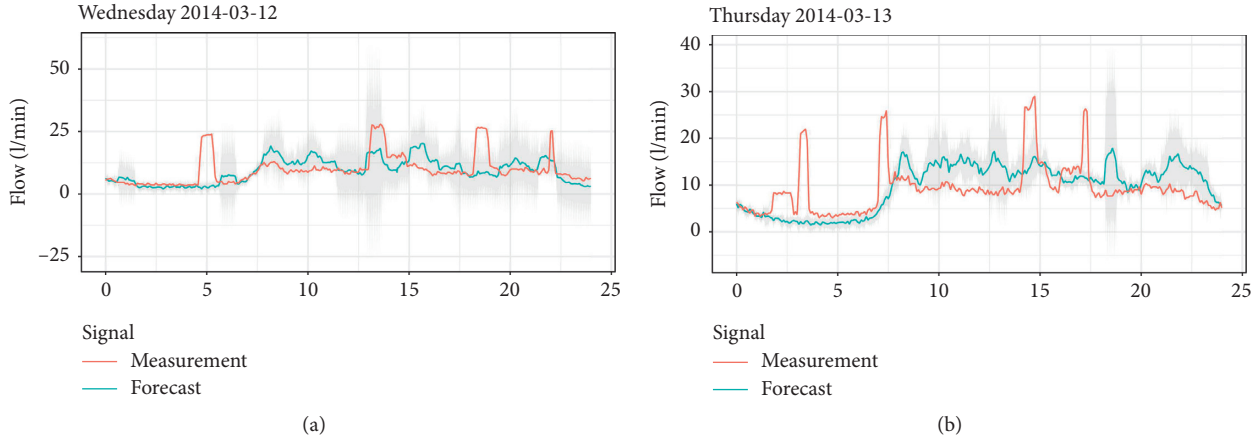


Figure 12: Comparison of two forecasts at the low-density population area site. The first one (a) was flagged as "green" in the FOB plot (FOB = 0.18), while the second (b) was flagged as "red" (FOB = 0.58). The pointwise 95% confidence band of the estimation is depicted in grey.

neighbors of the query pattern from the data in the history record.

As Dudek states [35], GRNN is a method equivalent to the STPS in terms of data preprocessing. It is also based on the X-Y pattern similarity that eliminates the seasonal components and also considers the nearest k neighbors. However, the prediction is made using a neural network of a single neuron in the intermediate layer using radial basis functions as weights. In this way, the result of the prediction is also an average of certain Y patterns, but with weights given by a Gaussian kernel (for further details, please refer [35]).

Other recently developed well-known approaches are based on ANNs, e.g., feedforward neural networks or long

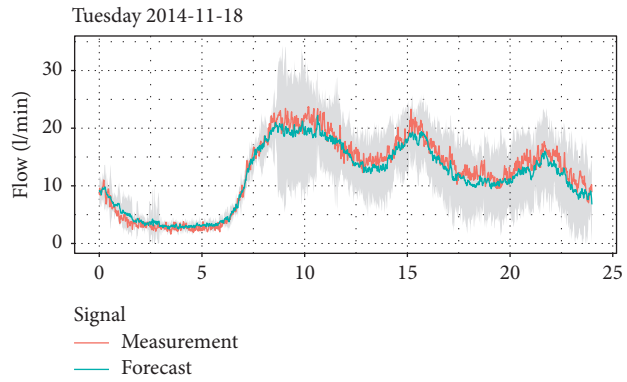


Figure 13: Example of a good forecast at the high-density population area site. The pointwise 95% confidence band of the estimation is depicted in grey.

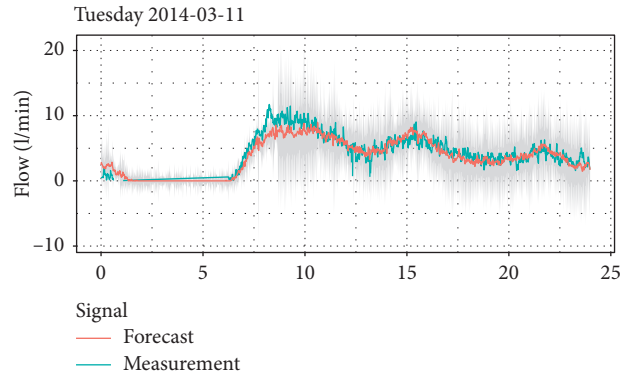


Figure 14: Example of a good forecast with missing data at the high-density population area site. There were missing/wrong data from approximately 1:00 a.m. until 6:00 a.m. Nevertheless, the overall outcome of the forecast was very accurate. The pointwise 95% confidence band of the estimation is depicted in grey.

short-term memory networks. Unfortunately, we could not apply these approaches to our study. From an engineering perspective, a 24 h forecast is commonly considered to be a short-term forecast; however, for data series with a high frequency, the number of predicted values is huge (1440 values). In our study, a method based on ANNs would require an output layer with 1440 nodes as well as an elevated number of nodes within both the hidden and input layers; therefore, the required data to train the ANN would be huge.

In the comparison study, we considered data from the year 2015 (in which the historical data is complete), and we computed the two error parameters proposed by Pacchin et al. [32]: MAPE and RMSE. Note that, since the number of data predicted for each day was high (1440 values) and because during an important fraction of the day, the water consumption values were very low (close to zero), and the value of the MAPE could become very high; therefore, for these special cases, we considered the RMSE to present a more adequate value to determine the goodness of the prediction.

In Figure 15, we present the measured values and the predicted values of the three approaches for a randomly-

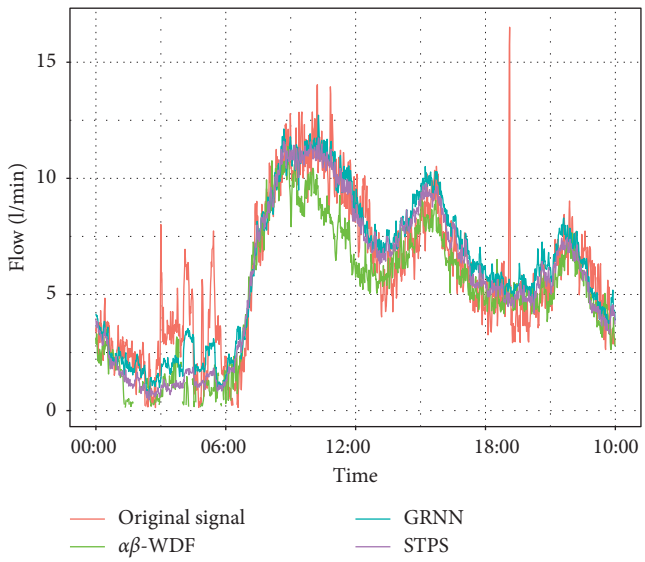


Figure 15: Forecasting at the high-density population area for Wednesday, 1 June 2015. The red line denotes the measured values, the green line denotes the predicted values by the $\alpha\beta$-WDF approach, the blue line denotes the prediction of GRNN, and the violet line denotes the predicted values by our approach (STPS).

picked day: 1 June 2015 at the high-density population area. Figure 16 illustrates the goodness of the methods under study in the three scenarios (industrial area, high-density population area, and low-density population area). It can be observed that our approach (STPS) and GRNN present better results than $\alpha\beta$-WDF for the three scenarios.

5. Conclusions and Future Works

This paper has presented an approach based on pattern-similarity techniques to forecast water demand. This work faces two important challenges that have been traditionally neglected in previous approaches, namely, a high frequency of predictions (based on measurements in terms of minutes) and the need for external data such as annual seasonality or weather that increments the complexity of the approaches. In that sense, on the one hand, the approach presented here is based on 1 min steps predictions, and, on the other hand, it does not require estimating annual seasonality since it determines this seasonality by constructing the X and Y patterns in which the series has been normalized.

In order to validate the approach, the study was applied over three different sites of a city in northern Spain. The results obtained provided interesting insights, such as the best predictions obtained in high-density population areas, the difficulties for identifying patterns for Sundays in industrial areas, or the higher random behaviour in low-density areas.

Additionally, our pattern-similarity approach (STPS) was also compared to other similar techniques that have been previously used for water forecasting, i.e., $\alpha\beta$-WDF and GRNN. The results obtained evidenced that $\alpha\beta$-WDF was the approach with worst results whilst GRNN and STPS behave similarly. This similar behaviour is normal since, in both cases, the estimation is obtained by an average

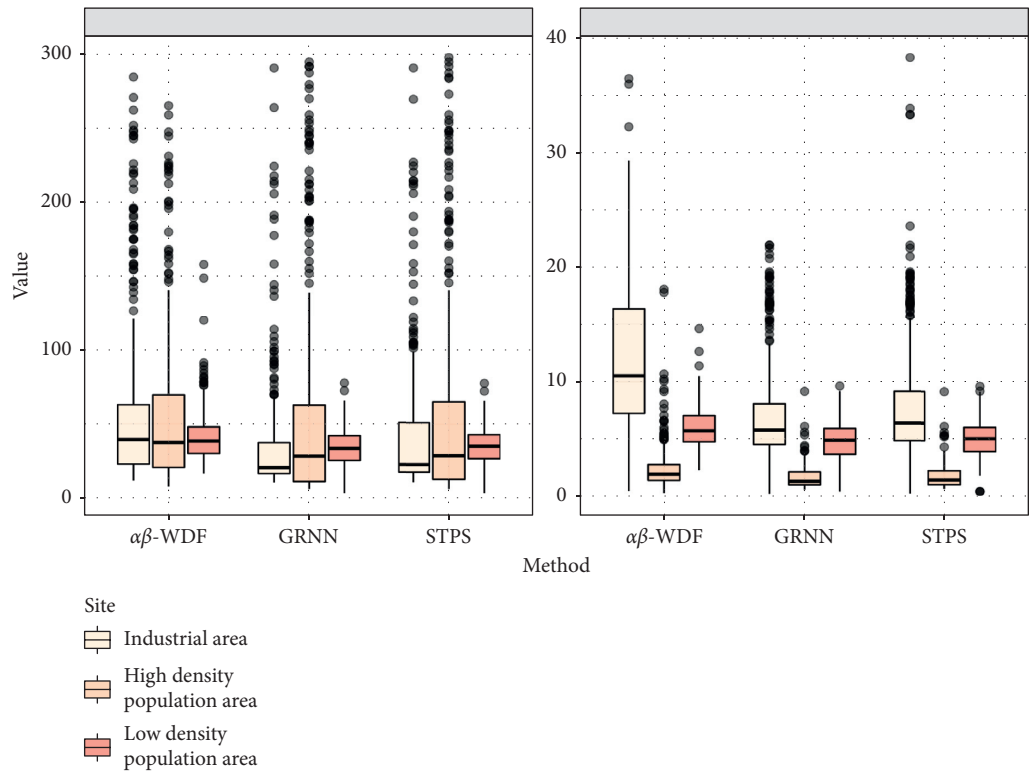


Figure 16: Goodness comparison among the $\alpha\beta$-WDF, GRNN, and STPS approaches at the three measured sites: (a) MAPE (%); (b) RMSE (l/min).

of some past values. However, while GRNN uses a weighted average where the weights are obtained by the Gaussian radial basis, in STPS, we use a simple average what is easier to compute. Thus, the main difference relies on the fact that, in GRNN, the abovementioned average is computed from some fixed previous days (one, two, or even three weeks before), while the STPS averages the last k nearest neighbors. Therefore, in cases in which the past weeks were, by any chance, nontypical (e.g., Christmas or Easter week), our method is providing better results due to its higher flexibility because in such cases it will look for similar days in the whole recorded history, whereas GRNN will be using only the past few weeks.

As future work, we intend to handle some weaknesses identified in the current method. Firstly, predictions success is reduced when anomalous days are considered. Anomalous days refer to two different situations: holidays and days with a behaviour different from what is considered usual. The former may be solved by constructing training sets from which to obtain the nearest neighbors since historical record contains enough data. To tackle the complexity of the latter, once these types of anomalous days have been identified, they could be just removed from the possible training subsets in the forecasting of other days.

Secondly, in order to improve the results for data sites where apparently there is not regularity, such as the low-density population area in our study, shorter prediction horizons could be considered, e.g., 4–6 hours. However, this is an issue that remains currently untested.

Finally, another interesting line of further work is the application of the proposed approach for water distribution in different cities.

Data Availability

The water consumption data of the Spanish city of Burgos, used to support the findings of this study, have not been made available because restrictions apply to the availability of these data, which were used under license for the current study and so are not publicly available. Data are however available from the authors upon reasonable request and with permission of the company Acciona Agua concretely through its subsidiary Aguas de Burgos.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors wish to acknowledge the collaborative funding support from (i) Ministerio de Ciencia, Innovación y Universidades (MCIU), Agencia Estatal de Investigación (AEI), and European Regional Development Fund (ERDF) Project (RTI2018-098652-B-I00); (ii) POCTEP 4IE Project (0045-4IE-4-P); and (iii) Consejería de Economía e Infraestructuras/Junta de Extremadura (Spain), European Regional Development Fund (ERDF) Projects (IB16055 and GR18112).

References

[1] R. Espinosa, L. Garriga, J. J. Zubcoff, and J. N. Mazón, "Linked open data mining for democratization of big data," in *Proceedings of the IEEE International Conference on Big Data (Big Data)*, pp. 17–19, Los Alamitos, CA, USA, 2014.

[2] A. Taivalsaari and T. Mikkonen, "A taxonomy of IoT client architectures," *IEEE Software*, vol. 35, no. 3, pp. 83–88, 2018.

[3] A. Degbelo, C. Granell, S. Trilles, D. Bhattacharya, S. Casteleyn, and C. Kray, "Opening up smart cities: citizen-centric challenges and opportunities from GIScience," *ISPRS International Journal of Geo-Information*, vol. 5, no. 2, p. 16, 2016.

[4] A. Di Nardo, M. Di Natale, G. F. Santonastaso, and S. Venticinque, "An automated tool for smart water network partitioning," *Water Resources Management*, vol. 27, no. 13, pp. 4493–4508, 2013.

[5] W-SMART Association, *Background and FP7 Goals*, W-SMART Association, Paris, France, 2018.

[6] Alliance for Water Efficiency, *Water Loss Control—Efficiency in the Water Utility Sector*, Alliance for Water Efficiency, Chicago, IL, USA, 2018, http://www.allianceforwaterefficiency.org/Water_Loss_Control_Introduction.aspx.

[7] A. D. Nardo, M. D. Natale, R. Gargano, C. Giudicianni, R. Greco, and G. F. Santonastaso, "Performance of partitioned water distribution networks under spatial-temporal variability of water demand," *Environmental Modelling & Software*, vol. 101, pp. 128–136, 2018.

[8] Center for Neighborhood Technology (CNT), *The Case for Fixing the Leaks: Protecting People and Saving Water while Supporting Economic Growth in the Great Lakes Region*, Center for Neighborhood Technology (CNT), Chicago, Illinois, USA, 2013.

[9] G. Dudek, "Pattern similarity-based methods for short-term load forecasting—part 1: Principles," *Applied Soft Computing*, vol. 37, pp. 277–287, 2015.

[10] G. Dudek, "Pattern similarity-based methods for short-term load forecasting—part 2: Models," *Applied Soft Computing*, vol. 36, pp. 422–441, 2015.

[11] G. Dudek, "Pattern-based local linear regression models for short-term load forecasting," *Electric Power Systems Research*, vol. 130, pp. 139–147, 2016.

[12] S. Meliá, J. Gómez, S. Pérez, and O. Díaz, "A model-driven development for GWT-based rich internet applications with OOH4RIA," in *Proceedings of the Eighth International Conference on Web Engineering, ICWE*, vol. 14–18, pp. 13–23, Yorktown Heights, New York, USA, July 2008.

[13] Y. Martínez, C. Cachero, and S. Meliá, "MDD vs. traditional software development: a practitioner's subjective perspective," *Information and Software Technology*, vol. 55, no. 2, pp. 189–200, 2013.

[14] R. J. Hyndman and G. Athanasopoulos, *Forecasting: Principles and Practice*, OTexts, Melbourne, Australia, 2nd edition, 2018.

[15] L. Shvartser, U. Shamir, and M. Feldman, "Forecasting hourly water demands by pattern recognition approach," *Journal of Water Resources Planning and Management*, vol. 119, no. 6, pp. 611–627, 1993.

[16] S. G. Buchberger and L. Wu, "Model for instantaneous residential water demands," *Journal of Hydraulic Engineering*, vol. 121, no. 3, pp. 232–246, 1995.

[17] S. G. Buchberger and G. J. Wells, "Intensity, duration, and frequency of residential water demands," *Journal of Water Resources Planning and Management*, vol. 122, no. 1, pp. 11–19, 1996.

[18] E. A. Donkor, T. A. Mazzuchi, R. Soyer, and J. Alan Roberson, "Urban water demand forecasting: review of methods and models," *Journal of Water Resources Planning and Management*, vol. 140, no. 2, pp. 146–159, 2014.

[19] M. Sebri, "Forecasting urban water demand: a meta-regression analysis," *Journal of Environmental Management*, vol. 183, pp. 777–785, 2016.

[20] J. Adamowski, H. Fung Chan, S. O. Prasher, B. Ozga-Zielinski, and A. Sliusarieva, "Comparison of multiple linear and nonlinear regression, autoregressive integrated moving average, artificial neural network, and wavelet artificial neural network methods for urban water demand forecasting in Montreal, Canada," *Water Resources Research*, vol. 48, no. 1, 2012.

[21] M. Herrera, L. Torgo, J. Izquierdo, and R. Pérez-García, "Predictive models for forecasting hourly urban water demand," *Journal of Hydrology*, vol. 387, no. 1-2, pp. 141–150, 2010.

[22] F. K. Odan and L. F. R. Reis, "Hybrid water demand forecasting model associating artificial neural network with fourier series," *Journal of Water Resources Planning and Management*, vol. 138, no. 3, pp. 245–256, 2012.

[23] G. Ji, J. Wang, Y. Ge, and H. Liu, "Urban water demand forecasting by LS-SVM with tuning based on elitist teaching-learning-based optimization," in *Proceedings of the Control and Decision Conference (2014 CCDC), the 26th Chinese*, pp. 3997–4002, IEEE, Changsha, China, May 2014.

[24] C. J. Hutton and Z. Kapelan, "A probabilistic methodology for quantifying, diagnosing and reducing model structural and predictive errors in short term water demand forecasting," *Environmental Modelling & Software*, vol. 66, pp. 87–97, 2015.

[25] A. Candelieri and F. Archetti, "Identifying typical urban water demand patterns for a reliable short-term forecasting—the icewater project approach," *Procedia Engineering*, vol. 89, pp. 1004–1012, 2014.

[26] A. Candelieri, D. Soldi, and F. Archetti, "Short-term forecasting of hourly water consumption by using automatic metering readers data," *Procedia Engineering*, vol. 119, pp. 844–853, 2015.

[27] A. Candelieri, "Clustering and support vector regression for water demand forecasting and anomaly detection," *Water*, vol. 9, no. 3, p. 224, 2017.

[28] S. Alvisi and M. Franchini, "Assessment of predictive uncertainty within the framework of water demand forecasting using the model conditional processor (MCP)," *Urban Water Journal*, vol. 14, no. 1, pp. 1–10, 2017.

[29] B. M. Brentan, E. Luvizotto Jr., M. Herrera, J. Izquierdo, and R. Pérez-García, "Hybrid regression model for near real-time urban water demand forecasting," *Journal of Computational and Applied Mathematics*, vol. 309, pp. 532–541, 2017.

[30] M. Romano and Z. Kapelan, "Adaptive water demand forecasting for near real-time management of smart water distribution systems," *Environmental Modelling & Software*, vol. 60, pp. 265–276, 2014.

[31] F. Gagliardi, S. Alvisi, Z. Kapelan, and M. Franchini, "A probabilistic short-term water demand forecasting model based on the Markov chain," *Water*, vol. 9, no. 7, p. 507, 2017.

[32] E. Pacchin, S. Alvisi, and M. Franchini, "A short-term water demand forecasting model using a moving window on previously observed data," *Water*, vol. 9, no. 3, p. 172, 2017.

[33] E. Arandia, A. Ba, B. Eck, and S. McKenna, "Tailoring seasonal time series models to forecast short-term water demand," *Journal of Water Resources Planning and Management*, vol. 142, no. 3, article 04015067, 2016.

[34] M. Bakker, J. H. G. Vreeburg, K. M. van Schagen, and L. C. Rietveld, "A fully adaptive forecasting model for short-term drinking water demand," *Environmental Modelling & Software*, vol. 48, pp. 141–151, 2013.

[35] G. Dudek, "Neural networks for pattern-based short-term load forecasting: a comparative study," *Neurocomputing*, vol. 205, pp. 64–74, 2016.

Hindawi
Scientific Programming
Volume 2019, Article ID 7973289, 21 pages
https://doi.org/10.1155/2019/7973289

Research Article

Practical Experiences in the Use of Pattern-Recognition Strategies to Transform Software Project Plans into Software Business Processes of Information Technology Companies

C. Arevalo,[1,2] **I. Ramos**,[1,2] **J. Gutiérrez**,[1,2] **and M. Cruz**[1]

[1]*Department of Computer Languages and Systems, University of Seville, Seville, Spain*
[2]*Web Engineering and Early Testing Research Group, University of Seville, Seville, Spain*

Correspondence should be addressed to C. Arevalo; carlosarevalo@us.es

Received 2 January 2019; Revised 18 February 2019; Accepted 26 February 2019; Published 2 May 2019

Guest Editor: Juan Carlos Preciado

Business process management (BPM) is a strategic advantage for all kinds of organizations, including information technology companies (ITCs), which cannot stay out of the BPM approach. ITCs manage business processes like projects to create and maintain software. Although Project Management Systems (PMSs), such as Microsoft™ Project Server® (MPS®), are considered as non-process-aware information systems (Non-PAISs), they may be a source to generate processes. In this paper, we propose a reverse engineering approach, which uses patterns to transform software projects stored in MPS® legacy databases into software business processes. For this, we base on the model-driven engineering paradigm and deal with the time perspective of the processes. This kind of experiences are scarce or almost nonexistent, so we show the AQUA-WS project case study, which runs with MPS® as source system and software process modeling languages as target systems. ITCs can benefit from this research by gathering knowledge about perspectives of their processes that would otherwise be wasted, such as executed projects or expired documents used in Non-PAISs. This fact can become a key factor for ITCs, which can increase their competitiveness and reduce software costs, as part of the BPM lifecycle of continuous improvement.

1. Introduction

Competitiveness in the global economy is one of the most important challenges that companies and organizations must face up. Therefore, business process management (BPM) is a strategic advantage that all companies should consider [1, 2]. Information technology companies (ITCs) focused on software process management are intensely involved in these challenges [3, 4], although their business processes are more complex, variable, and unpredictable [5, 6] than those that take place in other industrial sectors. However, no ITC can afford to reject the BPM approach. In turn, model-driven engineering (MDE [7]) has been promoted during the last decades as a paradigm to solve the complexity associated with processes of software management. Object Management Group (OMG) Model-Driven Architecture (MDA [8]) is the major exponent of MDE in the field of software engineering. Henceforth, we will refer to MDA in this article. ITCs have been working with this paradigm with certain level of success [9–12]. They often manage their operations using IT systems (ITSs) that may be classified, among others, in two categories: (i) process-aware information systems (PAISs) [2], where the concept of process is well defined, comprising models and traces of instance executions that are stored in an *event log*; and (ii) non-process-aware information systems (Non-PAISs), which are often Legacy Information Systems (LISs) [13–16], containing a Legacy Database (LDB) that stores states of ITC transactions.

The BPM lifecycle of continuous improvement [2] proposes analyzing process executions against current process models. Business process discovery (BPD), concerning process mining [17–19] in the scope of PAISs, executes algorithms to construct new models from traces of process instances that are stored in the *event log*. New and old models may be compared, thus enabling business experts

to optimize processes. At this point, we wonder what happens with BPD in the scope of Non-PAISs that lack the *event log*. According to van der Aalst [20], LDBs store a lot of hidden evidence or knowledge related to process execution, so that they may be good sources to extract *Process* dimensions, even in the case of Non-PAISs. Regarding BPD from Non-PAISs, some authors, such as Adam et al. [21] and Zou et al. [22], propose techniques to recover processes, whereas other researchers, such as Pérez-Castillo et al. [10, 23, 24], Arevalo [25], and Arevalo et al. [26–28], use an MDA-based approach called Process Archeology (PA) for that purpose.

Business processes include different dimensions or perspectives. Among them, *Control Flow* is the essential one, although there are other process perspectives, e.g., *Time*, *Organizational*, *Resources*, *Data*, and *Cases*. In this paper, we show an MDA-based approach to obtain software business processes of ITCs. For this, we initially point to the hidden *Time Dimension* that may be scattered in some databases (see work by van der Aalst [20]) that ITCs use to manage their software lifecycle. To demonstrate the suitability of our approach, we have developed a case study with a public company named EMASESA (http://www.emasesa.com). This company is responsible for the cycle of water supply and sanitation networks in the city of Seville. It has developed a big modernization software project called AQUA-WS [29], which manages the transformation of old client-server LISs into a new Web-based integrated system. With regard to this case study, we highlight the fact that this project has involved multiple companies and organizations to carry out the following challenge: "*To develop an integrated and modular software solution to manage EMASESA*." Basically, we have looked at the software process lifecycle used by different actors to develop the IT system of EMASESA. We do not focus on EMASESA business processes that run the cycle of water and sanitation networks, but to processes of software to solve its IT system. In this software project, we have played the role of main actors in the methodological field. The software lifecycle has been managed with NDTQ-Framework [30], following the Navigation Development Technique (NDT) by Escalona et al. [31]. In this case, NDT *Activities* are organized under a waterfall software lifecycle and Microsoft™ Project Server® (MPS®) has been the selected project management environment. We have applied our MDA-based approach to this case study in order to generate process models from source project plans that are stored in an MPS® database. This database suitably represents the *Time Dimension* of the project. For this reason, we have mainly faced up this dimension, although we are also interested in the others, since they may help to enrich process models. Besides, we have developed a specific Metamodel to extract *Time Dimension* from one or more source databases as well as to solve redundancy problems regarding *Activities* that are replicated in different LISs. We later discuss criteria for (i) selecting and classifying project tasks and (ii) mapping artifacts from projects onto process artifacts. Next, we analyze results, strengths, and weaknesses of processes. It must be mentioned that our approach can be applied in other software development projects involving ITCs that use MPS® to plan and control processes since it would be enough to use other project plans. Moreover, any other organization whose business processes are project-oriented and uses MPS® could be a candidate to benefit from the proposal, even if it is out of the software sector. They could obtain representations of instances of their processes and process models, if the instances have well-classified activities within categories or types of activities. The use of other databases would also be possible but at the expense of generating new metamodels of them that capture the essence of the execution of processes in the organization.

In summary, the main objective of this paper is to obtain software business processes related to ITCs project plans since BPM is a strategic approach whose scope is significantly broader than simple software project planning and control. This approach will assist business experts in the implementation of the BPM approach and will facilitate the enrichment of processes with the data that are included in software project plans. We aim to propose new methods of process discovery from databases of Non-PAISs since Process Mining [17–19] performs the same from the PAISs *event log*. Our approach takes into consideration other existing methods in the literature to obtain processes from Non-PAISs. However, they focus on evidences related to different perspectives of processes, initially based on their *Time Dimension* as the basis for the heuristics of process generation. We have contributed with a method and a set of tools to facilitate the work of the experts in the software business. They will be able to reuse the hidden knowledge about software business processes stored in databases, which would otherwise be forgotten or wasted. Additionally, it should be added that we expect better levels of efficiency and effectiveness when compared to the manual analysis of those processes.

The rest of the paper is organized as follows: Section 2 presents related work. Section 3 shows the main topics of our approach to model the *Time Dimension* of software projects. Section 4 shows an MDA-based proposal for taking out processes from databases of Non-PAISs taking into account different roadmaps; a specific MDA-based business process discovery roadmap is developed, which allows transforming project plans from MPS®. Section 5 analyzes how this roadmap behaves in a real case study, the AQUA-WS project, where results, advantages, and limitations of the approach are discussed. To finish, Section 6 states conclusions and future lines of research.

2. Related Works

Nowadays, ITCs are still working with LISs. Some LISs are PAISs, whereas most of them are Non-PAISs. Process Mining Techniques [17–19], are suitable to carry out business process discovery with *event logs* existing in PAISs, but they are not a choice to extract processes from Non-PAISs due to the lack of *event log* files. OMG Architecture-Driven Modernization (ADM) [32], which is an OMG's MDA-based [8] proposal for the modernization [13–16, 33] of LISs, includes reverse [34, 35] and forward engineering roadmaps from an old source system to a new target one.

ADM uses Abstract Syntax Tree Metamodels (ASTM) and Knowledge Discovery Metamodels (KDM) to extract knowledge from source systems. We are more interested in reverse [34, 35] engineering roadmaps that may help us to discover processes from source LIS artifacts, which may be (i) source code; (ii) graphical user interfaces (GUI); or (iii) databases. We have focused on the last one because databases are the most stable artifacts in a LIS. Since MDA and ADM are general standards that do not focus on process discovery field, we have needed to explore other literature regarding database reverse engineering as well as specific proposals with the aim to discover processes from Non-PAISs.

There is a lot of research work related to databases reverse engineering, but we have just selected some of them due to the likeness that they keep to our work. We would like to mention the work by Cleve et al. [36], who propose data reverse engineering using System Dependency Graphs (SDG) that analyze Data Manipulation Language (DML) sentences of Structured Query Language (SQL) that are scattered in the application code. They propose a new database schema by adding new candidate and foreign keys that may be inferred from SDGs related to the DML SQL embedded in the application code. Additionally, it is also remarkable Arevalo [25] and Arevalo et al. [26, 27] dealing with reverse engineering databases (i.e., relational tables, declarative constraints, and triggers) to define business Event-Condition-Action (ECA) rules over processes, expressed by means of Unified Modeling Language (UML [37]) and Object Constraint Language (OCL). Similarly, but not oriented to processes, it is relevant the work by Cosentino and Martínez [38], who also extract UML classes and OCL rules from tables and triggers. Finally, the proposal of Zanoni et al. [39] point to the evolution of software systems by pattern detection for conceptual schema recovery in data-intensive systems.

As we are interested in BPD field, we have selected some research works by Pérez-Castillo et al.: (i) those which propose Modernization Approach for Recovering Business Processes from Legacy Systems (MARBLE) [10, 23, 24] as a framework that extends the ADM standard and (ii) Pérez-Castillo et al. [33], who propose recovering Web Services from databases. These studies [10, 23, 24, 33] work as ADM, with KDM and ASTM, recommending different steps with KDM to discover business processes. The authors point to relational database DML sentences to propose new relational database schemas using ideas such as those included in the aforementioned work by Cleve et al. [36]. The BPD approach can generate business processes of different sizes and structures that may be characterized by connectivity, density, and separability of artifacts. Generated processes may recurrently present disadvantages regarding quality parameters such as comprehensibility and modifiability [40]. Process refactoring [40–42] includes techniques to write alternative process instances by adding, deleting, or redistributing existing process artifacts. Artifacts may be activities, gateways, events, or control flows. The refactorized process is a new process instance with the same semantic as the source process instance, which is generated by applying some rewriting rules. Refactoring processes quality is evaluated using artifact-based measurements. Caivano et al. [40] evaluate the process quality perceived by experts (human-perceived measures). They compare both, artifact-based and human-perceived quality measures to conclude that "Process refactoring is worthwhile so that humans reach better levels of comprehensibility and modifiability."

In order to manage the software lifecycle, business experts use General Process Modeling Languages (GPMLs) [43–45], such as Petri Nets, Business Process Execution Language (BPEL), Business Process Model and Notation (BPMN) [46], and Event-Driven Process Chain (EPC) or Unified Modeling Language Activity Diagrams (UML AD) [37], together with other specific Software Process Modeling Languages (SPMLs), such as Software and Systems Process Engineering Metamodel (SPEM [47–49]), Software Engineering Metamodel for Development Methodologies [50], and Essence-Kernel And Language For Software Engineering Methods (Essence [51, 52]). Besides, we have taken into account NDTQ-Framework [30], because it is used in our case study: AQUA-WS Project [29]. Bonnet et al. [53] consider BPMN as the process modeling leading standard between users and business experts, which is increasingly being utilized in the software field [43–45].

Although we are interested in different process dimensions, this paper particularly addresses the *Time Dimension* to generate software business processes for ITCs. Processes may be used by software experts with business process management systems (BPMSs) in a BPM lifecycle of continuous improvement. There are many approaches defining aspects regarding the *Time Dimension* of processes, but for the purpose of this study, we have selected those that model *Time Rules* on projects. Among other works, Flores and Sepúlveda [54] analyze *Time Rules* in project plans with the aim to depict them as BPMN processes. They propose time patterns, even though this solution overloads the main *Control Flow* of the process by adding a lot of imperative artifacts. Furthermore, Time-BPMN [55] is a clean and elegant proposal that extends the language with *Time Rules* by means of incorporating new decorators that are not still supported by BPMN 2.0 standard [46]. Cheikhrouhou et al. [56] extend the original Time-BPMN proposal by introducing new *Time Rules*.

According to the work by van der Aalst [20], databases hide knowledge related to processes, so that they may be good sources to extract process dimensions from databases of Non-PAISs. This paper constitutes a theoretical foundation that allows using databases as a source to construct *event logs* in systems that lack them, so that generated logs can feed Process Mining Techniques [17, 18]. Building on this framework [20], the work by González López De Murillas et al. [57] is an initiative that uses database Redo logs as a source artifact to generate an *event log*. The second initiative by González López De Murillas et al. [58] goes on in the same direction, i.e., they propose a metamodel and tools to connect databases with Process Mining [17–19].

The aforementioned works, regarding BPD from Non-PAISs, only generate some aspects of them. Nonetheless, as

results may appear to be poor in the eyes of business experts, they are not widespread. We expect that our proposal will be able to generate richer results, looking at the specific field of software lifecycle management and capturing dimensions (initially *Time Dimension*, but extendible to others such as *Organizational*, *Resources*, *Data*, and *Cases*) of processes that these Non-PAISs may hide. Taking into account the previous related work, we suggest a reverse engineering approach, composed of an MDA [8] infrastructure and heuristic methods, which allows turning projects stored in legacy databases into software business processes of an ITC. In comparison with some approaches cited above, which just use different transformation steps between ASTM and KDM, our heuristics do not use KDM and initially centers on *Time Dimension* of processes that may be scattered in databases of ITCs, which otherwise would be wasted for BPM purposes. We compare (cf. Table 1) our approach, initially tested with the AQUA-WS [29] project, with above initiatives that are close to process discovering.

Other authors' approaches, which use SDG [36], ASTM [10, 23, 24], or KDM [10, 23, 24], gather all kinds of Non-PAISs database artifacts that do not focus on process dimensions. The approaches that use SDG [36], ASTM [10, 23, 24], or KDM [10, 23, 24] collect all types of artifacts existing in databases of Non-PAISs, although they do not face the dimensions of the processes as a heuristic basis for generating them. There are few or almost nonexistent experimental cases that utilize these proposals to extract processes. Additionally, the results obtained do not go beyond deriving conceptual database schemes or poor approximations to real processes. In this paper, we extend our initial proposal [25–27] and show a detailed framework that addresses the selection of database artifacts that are closely related to process execution traces. This framework supports different roadmaps depending on source systems and selected target languages that software experts use to describe their processes. To the best of our knowledge, we have not found out approaches in the literature as our proposal, i.e., focusing on metamodels (cf. † in Table 1) concerning process dimensions that are scattered and hidden in legacy databases.

3. Time Dimension of Business Processes

ITCs as many other organizations are introducing the BPM approach for improving [43–45, 53] their software business processes, which means a key factor to become more competitive. As previously mentioned, processes have different perspectives or dimensions. The main one is represented by *Control Flow* of *Activities*, although the *Information* perspective may also be depicted as *Data Flows*. Besides, *Cases*, *Organizational*, *Resource*, and *Time Dimension* may also be represented in relation to business processes. In previous work [28], we principally focused on analyzing the *Time Dimension* of processes. As *Time Perspective* is concerned, a *Time Rule* is a subtype of *Business Rule* [59], which is well defined in turn by several authors, such as Ross [60], Wagner [61], and Cheikhrouhou et al. [56]. They suggest different classifications.

Orchestrations [46] are private business processes executed in an organization, but there are more complex ones, such as Choreographies [46] or Inter-Organizational Business Processes [56, 62]. In this paper, we have addressed software business processes executed by ITCs as Orchestrations. We have identified *Time Rules* classes that usually constrain a process and we have also developed an approach [28] that proposes a *Time Rule Taxonomy* [28] concerning Orchestrations, which is defined in a Process Metamodel (cf. Figure 1(a)). This Metamodel has a minimum set of classes to reach a good level of interoperability [43, 63] between GPMLs and SPMLs. Time rules [28] are defined as OCL constraints. Figure 1(b) depicts an example of *Time Rule: "The Start to Finish" Time Dependency (TD)* between a *Successor Activity* and a *Predecessor Activity*.

Classes of Metamodel [28] support the definition of *Processes*, which are composed of a set of *Activities*. *Subprocess* is a subtype of *Process*, and the *Activity* class is specialized with *Task* and *Milestone* subclasses. Furthermore, *TC* (*Time_Constraint* class) and *TD* (*Time_Dependency* class) may also be defined. Regarding attributes (i) *Subprocess* maybe Ad Hoc (*isAdHoc*), that means, its *Activities* run in parallel, without additional restrictions; (ii) *Activity* has *name, scheduled* (*startCPM* and *endCPM*) and *executed* (*start* and *end*) *events*; (iii) *Time_Constraint* class contains the attributes *TC type* (*tc_type*), maximum (*maxDur*), and minimum duration (*minDur*), constraints dates (*start_sch* and *end_sch*), attributes concerning *Absence Constraint* (*isInAbsence*, *startAbsence*, and *endAbsence*), and finally the number of loops (*looptimes*) that an *Activity* may run in terms of *Cardinality TCs*; and (iv) *Time_Dependency* class includes properties *TD* type (*td_type*), whether the *Activity* has an *Absence Dependency* with respect to *Predecessor Activity* (*isInAbsence* and, optionally, time interval [*startAbsence*, *endAbsence*]). Events of a *Successor Activity* may be constrained with a time lapse (*leadOrlag*) in terms of *Predecessor Activity* event, sometimes a *lead* (*leadOrlag* is negative) and in other cases a *lag* (*leadOrlag* is positive).

The taxonomy [28] includes the following elements: (i) *Time Constraints (TCs)*, which only affect an *Activity* in a *Process* and (ii) *TDs*, which involve rules between two *Activities*. Both of them regulate the *start* and *end events* of *Activities*. Since we initially pay attention to software projects carried out by ITCs, we have selected, from the referenced *Taxonomy* [28], *Time Rules* that may be found in most Project Management Systems (PMSs) (cf. Tables 2 and 3). We detail the rules they comprise as follows:

(i) *Time Constraints (TCs)* are classified into (a) *Duration* of *Activities*; (b) *Fixed* or *inflexible start* and *end events*; (c) *Flexible start* and *end events*; (d) *Cardinality*, which establishes constraints over the loop iterations and duration; and (e) *Absence Constraint*, which avoids the execution of an *Activity*.

(ii) *Time Dependencies (TDs)* involve *Predecessor and Successor Activities*. They can be classified as follows: (a) *Rules defined in Allen's Interval Algebra* [64] and (b) *Absence TD*, which avoids the execution of a *Successor Activity* depending on *Predecessor events*.

Table 1: Modernization initiatives of Legacy Information Systems (LISs).

Initiative	Source artifacts			Approaches used to discover hidden knowledge from LISs				Results				
	LDB	App code	GUI	SDG	ASTM	KDM	MM† concerning process dimensions	UML	OCL	ECA rules	LDB schema generation	Business process discovery
ADM [32]	☑	☑	☑					☑			☑	
Cleve et al. [36]	☑	☑		☑							☑	
Cosentino and Martínez [38]	☑							☑	☑		☑	
Pérez-Castillo et al. [10, 23, 24]	☑	☑	☑		☑	☑		☑			☑	☑
Arevalo et al. [26, 27]	☑				☑			☑	☑	☑	☑	☑
Arevalo [25]	☑				☑		☑	☑	☑		☑	☑
Our current approach	☑				☑		☑	☑	☑		☑	☑

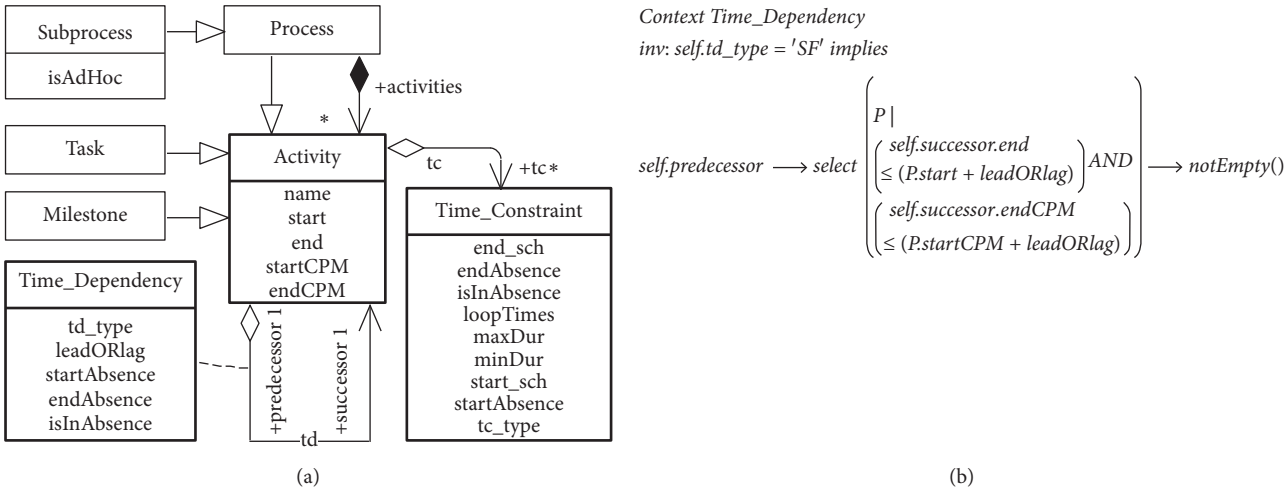


Figure 1: Process Metamodel [28] to capture Time Dimension. (a) Process Metamodel Class Diagram [28]. Project decomposition into Activity hierarchies and Time Rules: TCs and TDs. (b) An OCL sample TD included in Process Metamodel [28]; "Start to Finish "SF" Rule."

Table 2: TCs supported by PMSs.

Category	Time Constraint
Duration	FIXD: Fixed Duration FLEXD: Flexible Duration
Inflexible events†	MSO: Must Start On, MFO: Must Finish On
Flexible events†	ASAP: As Soon As Possible ⟶ S_{ASAP} and F_{ASAP} ALAP: As Last As Possible ⟶ S_{ALAP} and F_{ALAP} NET: Not Earlier Than ⟶ S_{NET} and F_{NET} NLT: Not Later Than ⟶ S_{NLT} and F_{NLT}

†S (start) or F (finish) events.

Table 3: TDs supported by PMSs.

Category	Time Dependency
Allen's algebra rules	SS: Start to Start SF: Start to Finish FS: Finish to Start FF: Finish to Finish

Arevalo et al. [28] include detailed definitions of each rule and OCL formulation.

4. An MDA-Based Approach to Generate Software Business Processes from Non-PAISs Legacy Databases Used by ITCs

ITCs are increasingly using the BPM approach to manage [43–45, 63] their software business processes, although they still use LISs that may be Non-PAISs as follows: (i) PMSs (such as MP®, MPS®, or Redmine®), which allow software experts to plan and control projects; (ii) Enterprise Content Management Systems (ECMs) (such as Alfresco® or Sharepoint®), which allow document management, collaboration, and subscriptions; (iii) a collection of ITs, e.g., Enterprise Resource Planning Systems (ERPs) (such as SAP® or Microsoft Axapta®), Customer Relationship Management Systems (CRMs) (such as Oracle Siebel®), and Supply Chain Management Systems (SCMs) (such as Kinaxis® or Blue Ridge®); and finally (iv) Tailor-Made Software

Systems. Furthermore, BPMSs are specialized ITs that may be integrated with other classic Non-PAISs. BPMSs support the BPM lifecycle of continuous improvement. Therefore, we have focused on how to reuse hidden knowledge of processes stored in databases, which would otherwise be forgotten and wasted.

This section shows our approach to generate software business processes from Non-PAISs legacy databases used by ITCs. Our proposal is an MDA-based framework that allows multiple reverse engineering roadmaps. Each roadmap implies a source system and a target system. We focus on (i) legacy databases as source systems and (ii) process modeling languages used by software experts as target systems. The first section establishes the architecture of our MDA-based solution. We show common aspects to all roadmaps. The second section focuses on ITCs as project-oriented organizations, who manage software project plans with MP®. We have developed a specific roadmap to transform software project plans stored in an MPS® legacy database into software business processes. Transformation heuristic is based on extensions of the Process Metamodel [28] (cf. Section 3). We will show detailed mapping rules with an algorithm and tables.

4.1. Architecture of Our MDA-Based Solution. We have pointed at specific processes of ITCs for software lifecycle management. Therefore, we have analyzed databases from diverse ITs (among others, MPS®, RedMine, Alfresco, and Sharepoint) gathering structures and rules concerning different dimensions of this kind of processes, because such databases hide a lot of knowledge generated by each ITC [20]. On these databases, we have studied the ability to extract structures and rules that are related to the main process dimensions, such as *Time*, *Resources*, and *Cases*. After the analysis, we come to the conclusion that (i) PMSs lay a strong foundation for *Time Dimension*, although they include definitions for *Resource* management; (ii) ECMs are suitable ITs for *Resource* management and also entail some *Time Rules*; and (iii) ERPs, CRMs, SCMs, and Tailor-Made software may involve rules concerning all process dimensions. Market or Standard systems are better choices than Tailor-Made ones since we can generate processes for many organizations that use the same system by utilizing the same roadmap. Initially, we have focused on PMSs, so we have analyzed *Time Rules* they support. Table 2 shows *TCs*, and Table 3 represents *TDs* that are usually included in PMSs.

Our approach is based on MDA [8] concepts. Figure 2 depicts a generalized MDA-based architecture to generate software business processes of ITCs from some databases of Non-PAISs. There may be different roadmaps to generate processes depending on the selected source and target systems. Each roadmap represents a concrete path that allows transformations from some source database artifacts into target Business Process Modeling Languages (BPMLs) that may be used to manage software business processes (GPMLs or SPMLS). The prerequisites for a candidate source database are (i) the source Non-PAIS (such as some PMSs, ECMs, ERPs, CRMs, or SCMs) must be used to manage software business processes and (ii) database must include some relevant artifacts (tables, constraints, and triggers) concerning the *Time Dimension* of software business processes of ITCs. A candidate database stores the hidden knowledge regarding processes of ITCs that we are looking for. Software experts choose their favorite SPMLs or GPMLs. With the aim of achieving greater interoperability, we propose to carry out the reverse engineering of processes up to our Metamodel [28], as an intermediate result that is platform independent, which means processes do not depend on the concrete syntax of any language. This Metamodel shares semantics (common classes and associations in process models) with the main SPML or GPML metamodels, which will allow us to easily export results through XML standard data exchange formats.

Our MDA [8] infrastructure consists of a set of metamodels at different levels of abstraction and transformations. Each model conforms to its metamodel, then metamodel rules are applied to each model. Transformations are based on heuristics in terms of our core Process Metamodel [28] (cf. Figure 1(a)). They offer interoperable models with concrete BPMLs. The approach could also be extended to capture other process dimensions from databases, such as *Resource*, *Organizational*, *Case*, or *Data*.

The main components of the MDA-based proposal are (i) source system, (ii) target system, and (iii) MDA [8] transformations.

(i) *Source System.* We have mainly looked at databases; therefore, it is important to know the data models that conform to *correspondent* metamodels. A Platform Specific Metamodel (PSM [8]) allows formalizing models within each source system. We must find database artifacts that are closely related to *Time Perspective* of processes, by means of analyzing reduced views that show task models, involving *Activities*, *Milestones*, and *Time Rules* gathered as hidden knowledge from source databases. These task models are represented on the technological platform corresponding to their Database Management System (DBMS), which is commonly a Relational Database Management System (RDBMS). That is why we need both generic metamodels (GASTM) and specific (SASTM) metamodels.

(ii) *Target System.* For software business processes of ITCs, the target system may be a BPML, either SPML (such as SPEM [47–49], ISO/IEC 24744 [50], Essence [51, 52], and NDTQ–Framework [30]) or GPML (such as BPMN [46] and UML AD [37]). These languages share some common characteristics along their process metamodels. They comprise the computer-independent (CIM [8]) level where ITCs business experts work.

(iii) *Heuristics to Generate Hidden Knowledge of Business Processes.* We propose a Model-To-Model (M2M) procedure [25, 27] that uses the previous MDA [8] infrastructure to explore databases. We have based

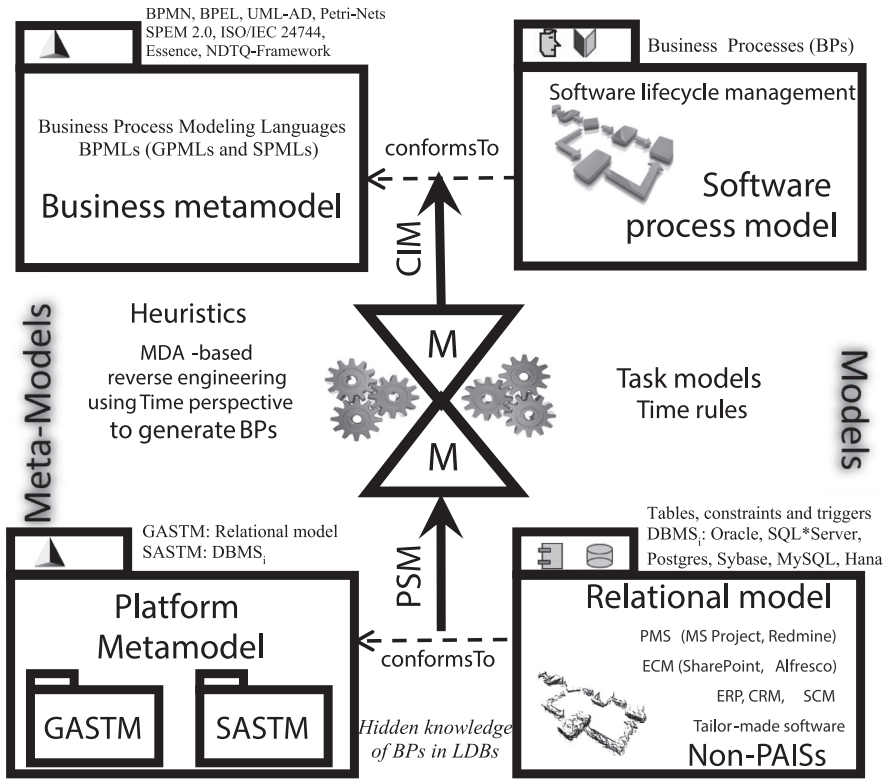


Figure 2: MDA-based infrastructure to generate software business processes (BPs) from LDBs used by ITCs. Candidate roadmaps to transform LDB task models into BPs.

the heuristics of process generation on identifying mapping among existing structures and rules in the PSM platform that correspond to classes and associations of our Process Metamodel to capture *Time Dimension* in a platform-independent metamodel (PIM [8]). This Metamodel has been proposed to extend GPML, such as BPMN, with time [28] (cf. Figure 1(a)) and also to support constraints of Tables 1 and 2, as well as some others that PMSs do not frequently use. The Metamodel has a minimum set of classes that share previous BPMLs (SPMLs and GPMLs); consequently, it is not difficult to match this point with the desired BPML.

At this point, we may also show a procedure (cf. Figure 3(a)) to generate software business processes from databases of ITCs using our MDA-based proposal. Figure 3(a) illustrates the general steps to follow:

(i) The activity "*To establish MDA generic infrastructure*" is executed as a subprocess that explodes in detail tasks in Figure 3(b), including "*To set a basic PIM Process Metamodel*" and "*To extend Metamodel with Time Rules*," which concern Process Metamodel [28] of Figure 1(a). If this proposal is used to generate processes from different database sources, we need a flexible and extensible *PIM Process Metamodel* that allows merging *Activities* and solving possible inconsistencies or redundancies. In our first case study, it has not been necessary to prepare and execute this activity yet. This first procedure is common to all possible roadmaps from a database of an ITC to a BPML.

(ii) Figure 3(c) depicts "*To Stablish MDA specific infrastructure depending on the roadmap*" into *Activities* "*To extract Task Process Metamodel from Source Legacy Database*" and "*To Extend Target Process Metamodel with PIM Metamodel Rules*". The former has to release database views including artifacts concerning business processes, their decomposition into *Activities*, and *Time Dimension* linked to such *Activities*. Prerequisites to use database of an ITC are associated with study database views (tables, attributes, constraints, and triggers; see Figure 4, which represents the database view in our case study) concerning hidden evidence of process executions; that is, it looks for instances of artifacts conforming to classes of *PIM Process Metamodel* [28] (cf. Figure 1(a): *Processes, Activities, Time,* and *Resource Rules*). The latter extends selected BPML with *Time Dimension* of *PIM Process Metamodel* [28] (Figure 1(a)). Both *Activities* are necessary for each specific roadmap. Efforts may be best monetized if selected source Non-PAISs and target BPMLs are widely used by ITCs to manage their software business processes. That is to say, a good roadmap goes from LISs with high market share to standard BPMLs that are well accepted by

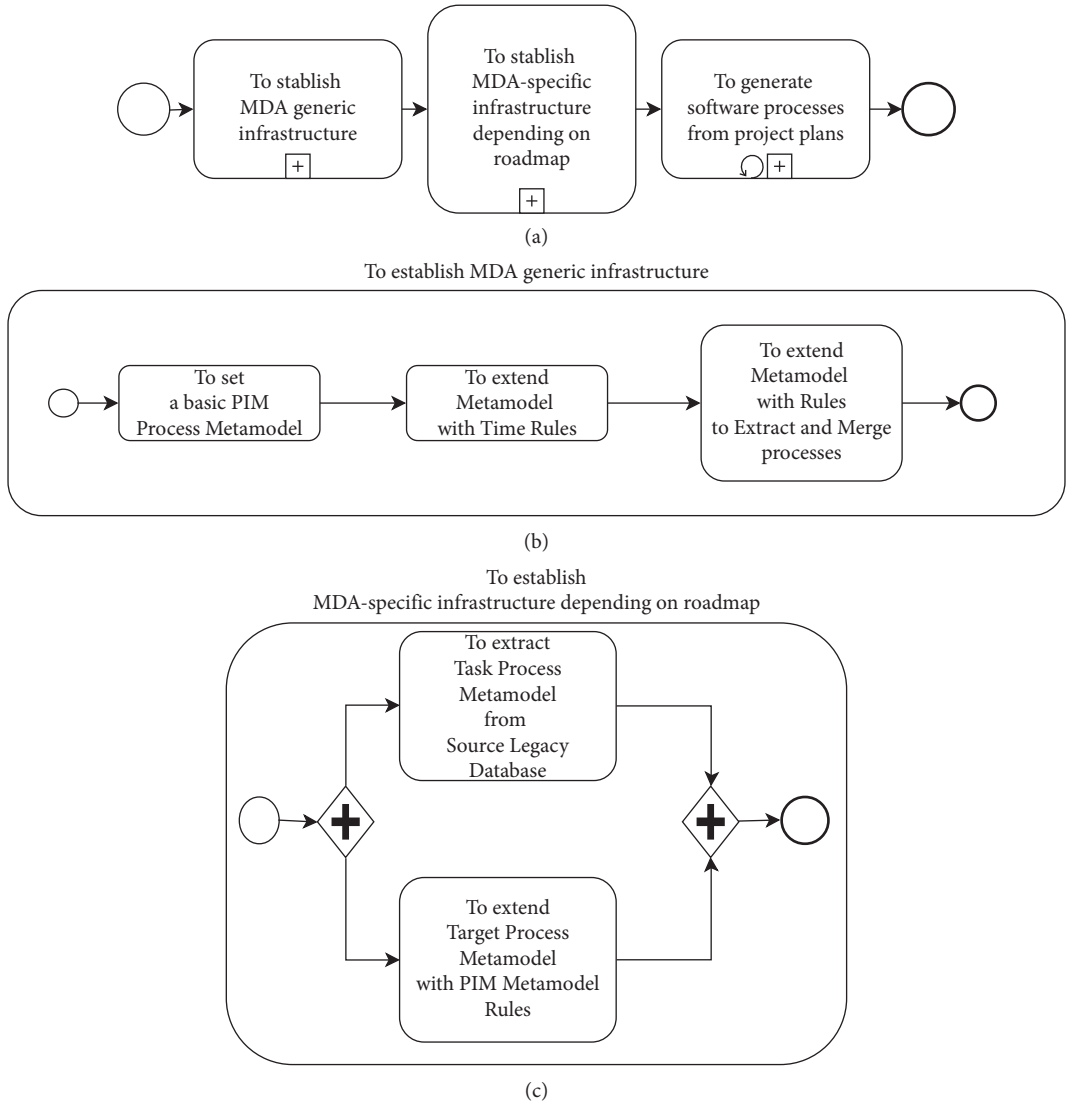


Figure 3: MDA-based procedure to transform projects stored in databases of PMSs into Software Business Processes of ITCs. It takes into account the common activities of all roadmaps and specific activities of a concrete roadmap. (a) Procedure to execute Software Business Process generation roadmaps. (b) Common activities of the MDA-based approach. Preparation of Process Metamodel [28] with Time Rules, as a Platform-Independent Metamodel (PIM) to execute all roadmaps. (c) Activities to execute each candidate roadmap, conditioned by source and target platforms.

business software experts so as to manage the software lifecycle.

(iii) Using this MDA infrastructure (common and specific to each roadmap), "*To Generate software business processes from project plans*" (Figure 3(a)) may be executed once or more times in a loop, in order to obtain software business processes from available sources.

4.2. An MDA-Based Roadmap to Transform Software Project Plans Managed with Microsoft Project into Software Business Processes. Regarding PMSs, Microsoft™ Project® (MP®) and Microsoft™ Project Server® (MPS®) are market products used throughout the world in organizations whose business processes are project-oriented; ITCs are not an exception, and MPS® has been used for many years by most of IT business experts all over the world, so we expect that our effort may be rewarded with the capability to generate a larger number of instances and models of processes available to these experts. Furthermore, we have analyzed RedMine and Alfresco, which are commonly used by ITCs, among other many other organizations out of the software field, so these source systems could also be in new roadmaps to generate processes of software. All of them are Non-PAISs whose project repository is stored in a legacy database. We have analyzed database metamodels of these products, but we focus on MPS® legacy databases, which are supported by four kinds of Microsoft™ SQL∗Server® instances: *Drafts, Published, Archive*, and *Reporting*. We have chosen *Published* instance, as it has the same structure as *Draft* instance, but stores the detailed information on tasks, links, and all

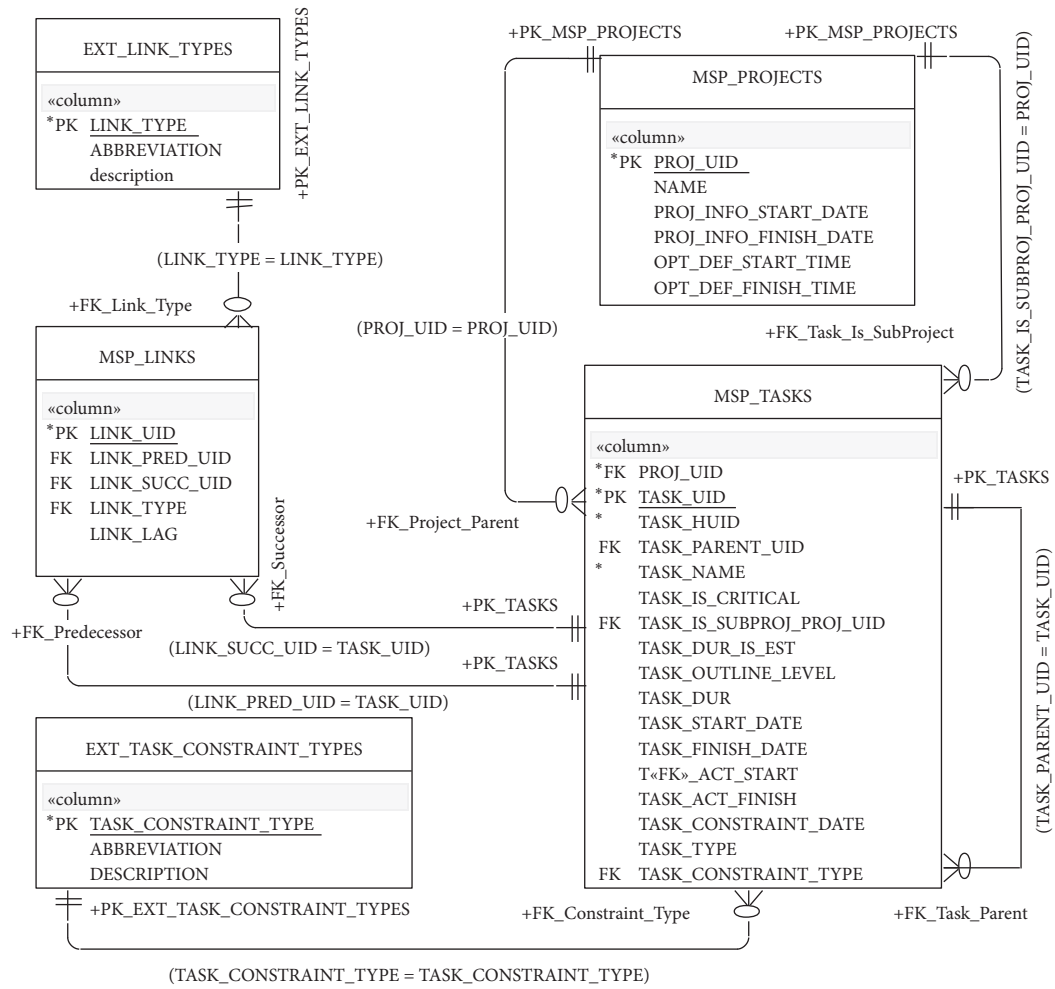


Figure 4: MPS® database task model (Published instance). Database view concerning *Time Dimension* of projects plans.

kind of constraints that the system provides for planning and replanning projects. Published and Archive instances are derived from Drafts and Published ones.

We have developed an initial roadmap shown in Figure 5 that is a specialization of Figure 2 (suitable for all roadmaps). MPS® legacy database is the selected source system. As experts choose their favorites software process languages, we do not want to constraint to only one language to describe software business processes, so the target system that will be our platform-independent Process Metamodel [28]. This way, it will be possible to use different BPMLs as targets, because artifacts (classes and rules) of our Metamodel exist in most of BPML metamodels commonly used by software experts. Generated process models can be represented with SPMLs or GPMLs that are widely used by ITCs. Powerful target systems are BPMN and the Workflow Management Coalition (WfMC) XML Process Definition Language (XPDL). They will allow the use of the World Wide Web Consortium (W3C) Extensible Markup Language standard (XML) to exchange serialized processes (schemas and instances). BPMN allow serializations with .xsd and .xmi XML formats and XPDL with .xpdl format. Most of BPMLs (GPML and SPML) supporting tools allow the use of .xsd, .xmi and .xpdl exchange formats to serialize processes. In summary, we have aimed to generate instances and process models from project plans focusing on task structures and Time Rules. As future work, we will be able to use further source systems, such as other PMSs, ECMs, ERPs, CRMs, SCMs, or Tailor-Made software. Besides, we will propose roadmaps to specific SPMLs or GPMLs targets, used by ITCs that work with this type of systems. In this case study, we have considered the following aspects of source systems, target systems, and heuristics to generate business processes:

(i) *Source System MPS®*. Database prerequisites to be a source for generating software business processes (cf. Section 4) involve exploring a metamodel regarding projects structure and their *Time Dimension*. Database Task model in Figure 4 allows us to explore a lot of projects in many ITCs that manage the software lifecycle. The source metamodel must support SQL∗Server, which conforms to ISO SQL 1992, so a generic relational metamodel (GASTM) for this standard and a specific metamodel (SASTM) for *Microsoft*™ SQL∗Server® [26] are needed. Task model (Figure 4) is taken out from

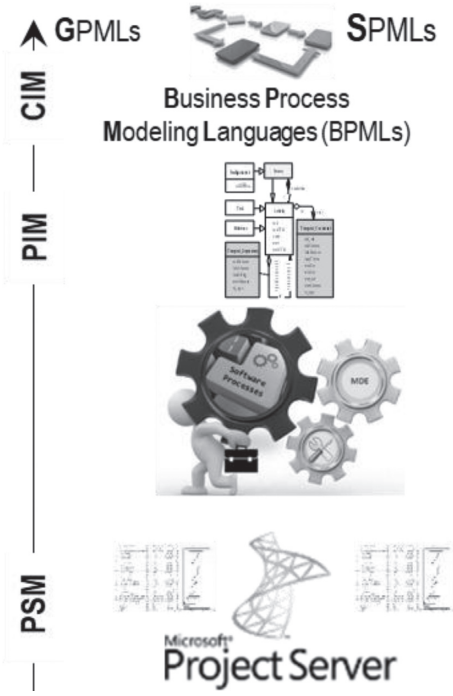


Figure 5: MDA-based roadmap from MPS® database to PIM Process MM [28] (as a roadmap case of Figure 2).

the MPS® Published instance. Relational *tables* and *foreign keys* are displayed below:

(a) *MSP_PROJECTS Table.* It stores information concerning projects.
(b) *MSP_TASKS Table.* It displays rows that show subordinated tasks of a project through the *FK_Project_Parent* foreign key. Moreover, if a task is a subproject, then it may be scheduled as an external project (*FK_Task_Is_Subproject*). Consequently, *Activities* may be organized hierarchically with task groups as parent tasks and child tasks. These relationships are expressed through *FK_Task _Parent.* The task table enables defining due dates over task *events* (*start* and *end*) as well as setting task duration (fixed or estimated). *TCs* of Table 2 are supported. *EXT_TASK_CONSTRAINT TYPES* table constitutes the enumeration of *TCs.*
(c) *MSP_LINKS Table.* It let us identify relationships between *Predecessor* and *Successor Tasks. TDs* of Table 3 are supported. *EXT_LINK_ TYPES* table represents the enumeration of *TDs.*

(ii) *Target System.* We have used our metamodel [28] (cf. Figure 1(a)), which holds extended *Time Rule's* semantics that has not been supported by languages such as BPMN yet. This target system works as a platform independent of technology. Gantt or PERT diagrams just allow planning or executing projects. BPMLs are increasingly used for modeling software business processes. BPMLs, such as SPMLs [43, 63] and GPMLs [43–45], are quite more powerful for modeling other dimensions, such as *Control Flow, Organizational, Data, and Resources*, as well as for being the gateway not only to model but to execute and audit those processes in a cycle of continuous improvement. From the point of view of software lifecycle management, focusing on processes gives the expert a much broader scope than just pointing to planning projects.

(iii) *MDA-Based Heuristics to Generate Business Processes.* The heuristics highlight the identification of *Time Rules* concerning projects (*TCs* and *TDs*). In this paper, we have solved mappings between the source system and target Time-based Process Metamodel [28], which are M2M transformations. Table 4 shows correspondences between the properties of the source system tables (project plans) and the attributes of the metamodel classes of processes with temporary rules. They aim to extract business processes regardless of the platform or target language:

(a) Table 4(a) depicts mappings of a *Project* onto a *Business Process.* MPS® allows to manage project hierarchies; this means that a *parent project* may have an activity that also is a *subproject.* In this case, there are two instances of projects, the parent one and the child subproject so a link for association "*is_a_Activity*" exists.
(b) Table 4(b) contains the details of mapping *project tasks* onto *Process Activities (+activities). Single tasks (activity is_a Task* or *is_a_Milestone)* are mapped as *Activity subtype.* Another activities hierarchies are "*Nested Tasks*" that means a *parent activity* is decomposed into *child activities.* If a project *task* were decomposed into more detailed tasks, then the corresponding *Activity* would be also a *Subprocess* and a *Subprocess* would be a subtype of a *Process.* These generalization relationships are solved in Table 4(c). *Subprocesses* are *AdHoc*, meaning that the execution flow runs in parallel for the group of tasks, without additional restrictions.
(c) Table 4(d) includes mapping rules for those cases where a task is an external subproject. It is mapped as a subprocess.
(d) Mappings of *Duration Rules* allowed by the source system are detailed in Table 4(e). *Fixed* (*FIXD*) and *Flexible Duration* (*FLEXD*) rules are expressed by OCL constraints.
(e) Table 4(f) shows mapping criteria for *TCs* over *Activity events*, including *Flexible* and *Fixed* rules concerning the *start* and *end* of an *Activity* (*TCs* of Table 2 are supported).
(f) Finally, Table 4(g) depicts mappings for *TDs* between a *Predecessor* and a *Successor Activity* (*TDs* of Table 3 are supported).
(g) Table 4(e)–4(g) include the column "*OCL constraints*," concerning *TCs* and *TDs*, thus generated processes inherit *Time Rules* that are

Table 4: Mappings from the source system onto the target system. Source: MPS® legacy database Metamodel (cf. Figure 4) ⟶ Target: Process Metamodel [28] (cf. Figure 1(a)).

Concept	Source MM		Target MM			
	Artifact	Columns	Class	Property	Associations	OCL constraints
(a) A Project is mapped onto a Process						
#Project Project name	Msp_Projects	proj_uid name	Process Activity	oid_Process name	is_a Activity	
(b) Project Tasks are mapped onto Process Activities						
#Activity Activity name Activity start Activity end	Msp_Tasks + FK_Msp_Projects	task_uid task_name task_act_start task_act_end	Activity	oid_Activity Name Start End	+activities (∗) is_a Task is_a_Milestone	
(c) Group Activities are mapped onto Subprocesses						
Activity is composed of a group of tasks	Msp_Tasks: Table + FK_Task_Parent	Parent of the set_of_rows with the same foreign key (Task_Parent_Uid = Task_uid) and (Msp_Projects.Uid = Msp_Tasks.Proj_Uid)	Activity Subprocess	Is_AdHoc = true	is_a Subprocess is_a Process	
(d) Subprojects are mapped onto Subprocesses						
Activity is an external subproject	Msp_Tasks: Table + FK_Task_Is_SubProject	Parent of the set_of_rows with the same foreign key Task_Is_SubProj_Proj_Uid = Proj_uid	Activity Subprocess		is_a Subprocess is_a Process	
(e) Allowed Durations are mapped onto OCL constraints						
Fixed duration Flexible duration	Msp_Tasks: Table	task_dur_is_est = false task_dur_is_est = true	Time_Constraint	mindur = maxdur mindur = task_dur	tc(∗)	FIXD FLEXD
(f) Time Constraints are mapped over Activity events						
Inflexible TC Flexible TC	Msp_Tasks: Table	task_dur_is_est = false task_dur_is_est = true	Time_Constraint	mindur = maxdur mindur = task_dur	tc(∗)	FIXD FLEXD
(g) Time Dependencies are mapped over Activity events						
Time Dependencies	Msp_Links FK_Link_Type FK_Predecessor FK_Successor	Link_type link_lag	Time_Dependency	td_type = link_type leadORlag = link_lag	+predecessor +successor	SS, SF, FS, FF

defined in PIM Process Metamodel [28] as OCL invariants and derivation rules.

Algorithm 1 shows the procedure to carry out M2M transformations whose mapping details are shown in Table 4.

5. Generating Software Business Processes by Running the Roadmap from Source MS Project Software Plans: The AQUA-WS Project Case Study

AQUA-WS Project [29] is a multiyear software modernization project leading a new Web-based system by means of multiple and heterogeneous old client-server legacy systems (cf. Figure 6, which depicts four subsystems composed of sixteen applications for water lifecycle management and infrastructures).

The project has been carried out for EMASESA, which is a public company in the utility sector that runs the water lifecycle in the city of Seville. It has been developed by international software companies in liaison with some research groups, from the University of Seville and the University of Malaga, amounting an investment of 3.5 million euros. Our research group, Web Engineering and Early Testing (http://iwt2.org), has been responsible for the work in methodological support and quality assurance of the project. We want to highlight that we have chosen this case not for the management of the water cycle that is the business of EMASESA but for the management of the software development process that has involved all the actors in this significant project.

The AQUA-WS case study [29] has allowed us to validate the proposed MDA-based framework. It has also helped us to extend the proposal to generate and merge processes from different LISs including more perspectives, such as *Organizational*, *Resources*, *Data*, and *Case* dimensions. In light of this, we describe the environment of the study in Section 5.1.

5.1. Environment of the Study. All the teams have used either stand-alone clients or Web UI interfaces of MPS® to face up their work. This software collects and centralizes all the information related to the development of different subsystems in those cases where the responsibility is shared among teams. AQUA-WS is just a case of the MDA-based initial roadmap developed in Section 4.2.

NDT methodology [31] has been the reference used to manage the AQUA-WS Project, and NDTQ–Framework [30] has offered facilities to support NDT and automatically generate documentation in an MDE-based project. NDTQ-Framework is implemented with Sparx Systems Enterprise Architect, which has helped us to customize M2M transformations by means of OMG Query/View/Transformation (QVT) [65] language and plugins. The project has been organized with different teams: (i) development team of each software company; (ii) quality assurance team; and (iii) customer team.

5.2. Analysis of Results. In this section, we analyze results obtained by means of the approach to generate software business processes from MPS® AQUA-WS legacy database, as the source LDB. Table 5 summarizes the transformations from units of the source project plan into target process elements. Figure 7 is the source Gantt chart regarding the schedule of software development *Activities* of AQUA-WS project [29], and Figures 8–10 represent processes generated by the MDA-based approach.

Processes are depicted as BPMN diagrams although it is easy to show them in other languages selected by software experts, such as SPEM, which is more appropriate to deal with this type of business processes. It is easy to change the final step of the reverse engineering procedure, which transform our PIM Metamodel [28] into the software process modeling language. This is because our PIM Metamodel [28] has a minimum number of classes and associations that always exist in metamodels of these software process modeling languages. Particularly, the transformation to the BPMN Metamodel also allows us to export results to other metamodels of software process modeling languages by using XML standard serialization formats such as .xsd and .xmi. These standards facilitate the interoperability between technological environments to exchange the obtained processes.

We have gathered three kinds of *activities* to apply the proposed MDA-based process generation (cf. Figure 2, which depicts most of the possible roadmaps and Figure 4, which illustrates the specific roadmap that is solved in this paper). It uses metamodels (Figures 1(a) and 4) and mapping rules (Algorithm 1 and Table 4) that we have previously shown. Each category of *activities* and related processes are described as follows:

(i) *Organization, Quality Assurance, and Subsystems Decomposition*. This level is a composition of general *Activities* either for organizational purpose or for specific quality assurance work (cf. Figure 8), together with a decomposition into AQUA-WS subsystems (cf. Figure 6), where each subsystem is allocated to the main development team.

(ii) *Development of Subsystems*. We have selected a subsystem, such as "Activity #61 (Alfa 0.4) Customers: Networks Intervention Subsystem" (cf. Figure 9: "Alfa 0.4. Customers. Networks Intervention Subsystem") to display its generated business process. Subsystems involve processes linked to some *Activities* included in NDT waterfall software lifecycle [31] (cf. Figure 11), from the requirement to implementation phases.

(iii) *NDT Phases*. Each NDT phase consists of *Activities* that may be optional or mandatory. For example, Figure 12 represents the corresponding process for the SA NDT phase, which, in this case, has been manually designed by the business software expert. Project manager uses patterns of NDT phases that are stored within the MPS® database. For instance, "Activity 1001" is an

Input: *J* project, which is stored in MPS® Database Published instance
Output: BP Process, which conforms to platform-independent (PIM) Process Metamodel [28] that includes OCL *Time Rules*
BP ⟵ new(Process); —*They refer to mappings included in Table 4(a)*
for each ((*TMP: Msp_tasks*) ∈ (*J: Msp_Projects*)){ —*It maps MPS® Tasks onto PIM MM Activities*
  A ⟵ *create_Activity* (*TMP*); —*It creates A: new Activity* ∈ BP *for each project task*
  create_TCs (*A, TMP*); —*It creates TCs: duration, fixed or flexible events*
  create_TDs (*A, TMP*); —*It creates TDs: dependencies between current task and their predecessors*
};
return BP;
function *create_Activity* (*TMP: Msp_Task*) { —*It creates an Activity and its corresponding subclasses*
  A ⟵ new(*Activity*) ∈ BP; —*The activity is included into* BP *(mappings included in Table 4(b))*
  case (*TMP*) {
    "*Subproject*" **or** "*Task_Group*": {
      P ⟵ *new*(*Process*) ▷ *A*; —*Subprojects/Groups are mapped as subprocesses*
      SP ⟵ *new*(*Subprocess*) ▷ *P*; SP ▷ *A*; —*It sets the hierarchy of Activities and subprocesses (Table 4(d))*
      if (*TMP* == "*Task_Group*") *SP.isAdHoc* := *true*; —*It groups tasks as AdHoc subprocesses (Table 4(c))*
    }
  "*Single Task*": *T* ⟵ *new*(*Task*) ▷ *A*; —*Activity is a single task*
  "*Milestone*": *M* ⟵ *new*(*Milestone*) ▷ *A*; —*It is a milestone*
  }
};
function *create_TCs* (*A: Activity,TMP: Msp_Task*) { —*It creates TCs: duration and time events*
  TC ⟵ new(*Time_Constraint*) ∈ *A*; —*It creates TC for activity duration (Table 4(e))*
  if (*TMP.task_dur_is_est*) **then** {*TC.tc_type* ⟵ "*FLEXD*"; —*It refers to estimated duration as Flexible Duration*}
  else {*TC.tc_type* ⟵ "*FIXD*"; —*It is fixed duration*};
  if (*TMP.task_constraint_type* ∈ {"*MSON*", "*SASAP*", "*SALAP*", "*SNET*", "*SNLT*", "*MFON*", "*FASAP*", "*FALAP*", "*FNET*", "*FNLT*"})
then {
    TC ⟵ new(*Time_Constraint*) ∈ *A*; —*It creates TC for scheduled end events (Table 4(f))*
  };
};
function *create_TDs* (*A: Activity,TMP: Msp_Task*) { —*It creates TDs* {"*SS*", "*SF*", "*FS*", "*FF*"}
  for each (*LK: Msp_links*) ∈ (*TMP: Msp_Task*) { —*It maps links onto dependencies*
    TD ⟵ *new*(*Time_Dependency*) ∈ *A*; —*It creates TD for scheduled events (Table 4(g))*
  }
};

Algorithm 1: PSM_MSProjectDatabase_TO_PIM_process_MM (*J*: *Msp_Projects*).

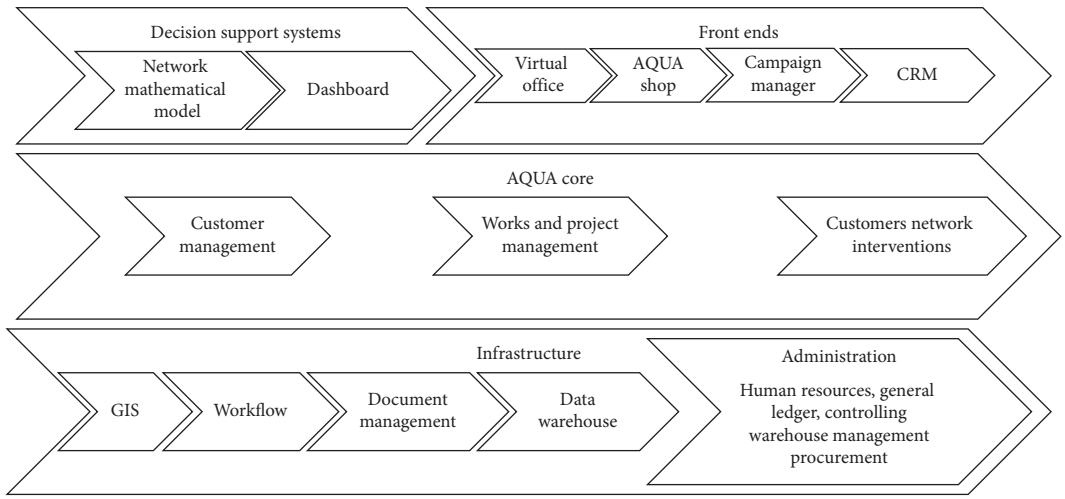


Figure 6: Information System Architecture of Emasesa, which is the object of AQUA-WS software development project; subsystems and applications to manage the cycle of water supply and sanitation networks.

Table 5: Mappings from source "*Software Project Plans*" onto target "*Software Business Processes*".

Source scheduled tasks		Target process elements	
Group of tasks	No. of tasks (Figure 7)	Process activities generated	Figures
Organizational Activities and Subsystems	#(1, 3) #(11, 40, 56, 61, 80, 94)	Planning and control together with quality assurance activities; parallel development of subsystems.	Figure 8
Decomposition of global task (Alfa 0.4) Customers: Networks Intervention into detailed tasks	#61 decomposed into #(62–69)	Process that represents a decomposition of a software subsystem into development phases. Phases that concern NDT methodology, which is applied in the context of software waterfall development lifecycle (Figure 11).	Figure 9
External subproject activities linked to a System Analysis (SA) NDT phase	#64	Decomposition of a concrete SA NDT phase associated with the application included in the subsystem #61 and related to SA NDT phase. Four models are generated; therefore, they may be compared to the NDT SA pattern phase (Figure 12).	Figure 10

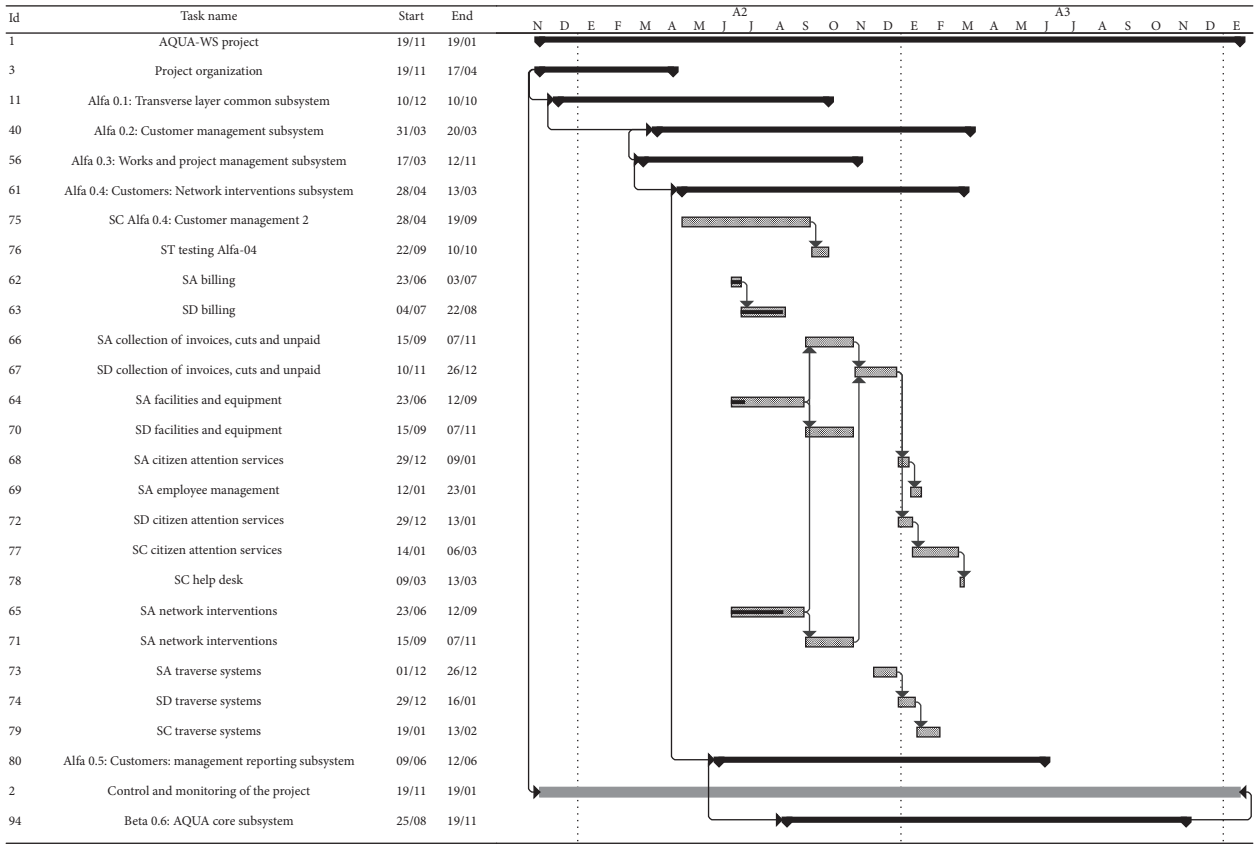


Figure 7: AQUA-WS project schedule. Organization and quality assurance activities, subsystems scheduling, and detailed NDT phases of subsystem "Alfa 0.4".

external subproject for NDT SA phase that may be linked to a specific SA *Activity* contained in a subsystem. Later, *Activities* inherited from the pattern may be modified in their corresponding subsystems. We have chosen "Activity #64 SA of Facilities and Equipment Subsystem" (cf. Figure 10) in this third category.

We comment on some aspects of process generation as follows:

(i) To start, *Project Tasks* are mapped onto *Process Activities* and *Control Flow* in each piece of the process appears as a consequence of *TDs*. On the one hand, FS *TD* induces sequential flow (⟐); on the other hand, SS, SF, and FF *TDs* are represented with parallel flows (⟐), without additional restrictions.

(ii) Next, *Hierarchies of Activities* are solved as follows: (a) *Parent Activity* is mapped onto an *AdHoc Subprocess*; (b) *Child Activities* run in parallel.

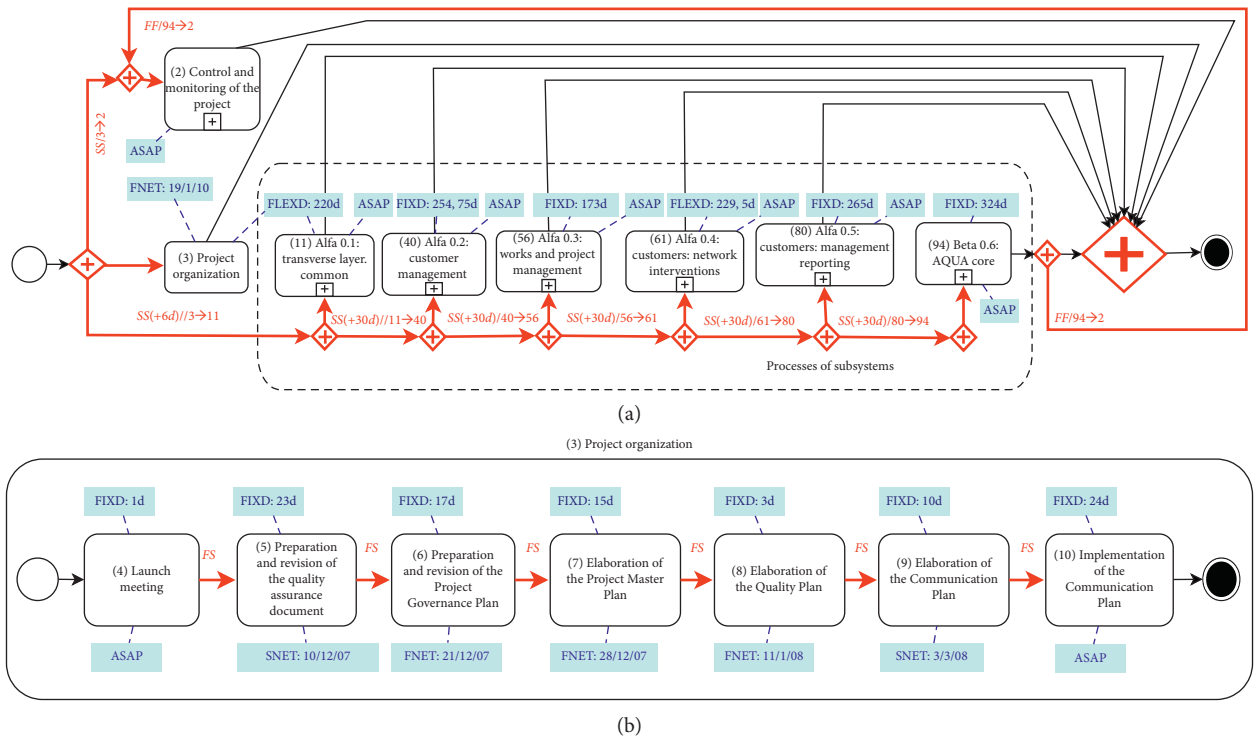


Figure 8: Generated process for MPS® "Organizational Activities and Subsystems". Control Flow of activities and Time Rules (TCs and TDs) extracted from the project plan.

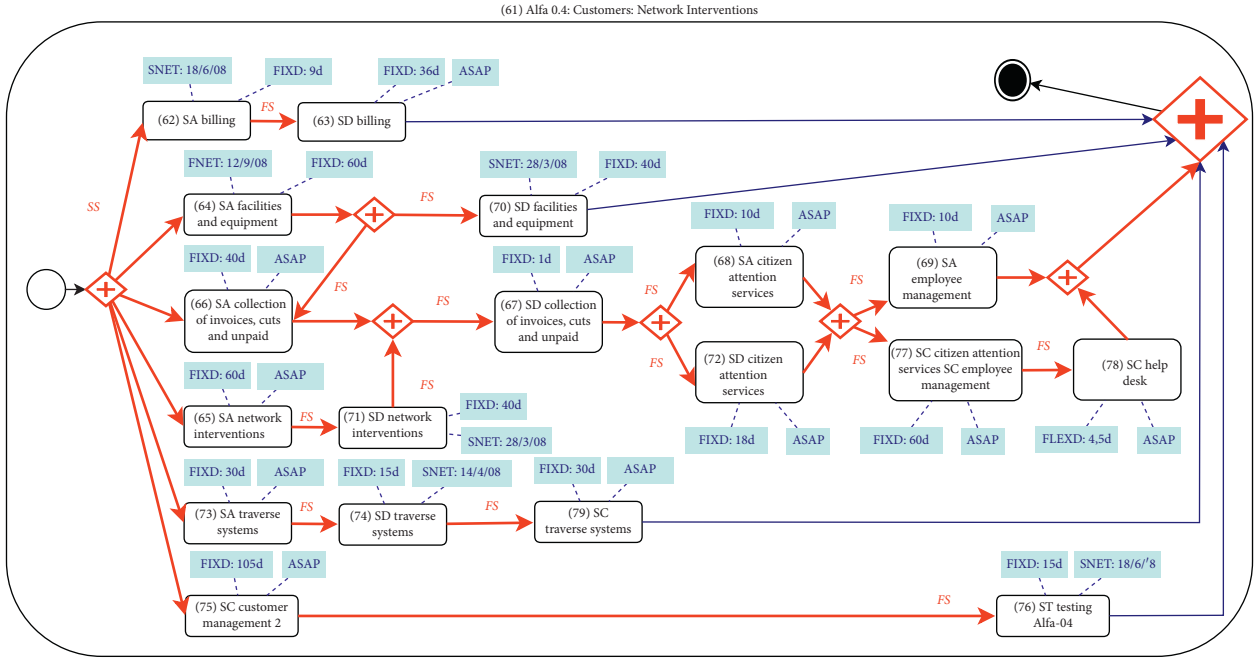


Figure 9: Generated process: "Activity #61 (Alfa 0.4) Customers: Networks Intervention Subsystem." Control Flow of Activities and Time Rules (TCs and TDs) extracted from the project plan.

Attending to time semantics generated processes enforce the selected *Time Rules* of our Metamodel [28] (cf. Figure 1(a)), which are expressed by OCL assertions [28] through our MDA-based approach.

(iii) To finish, *TCs* are allocated to their corresponding *Activities*, enriching each generated process. We can assure that all tasks in a project have been properly captured and grouped into *Subprocesses*

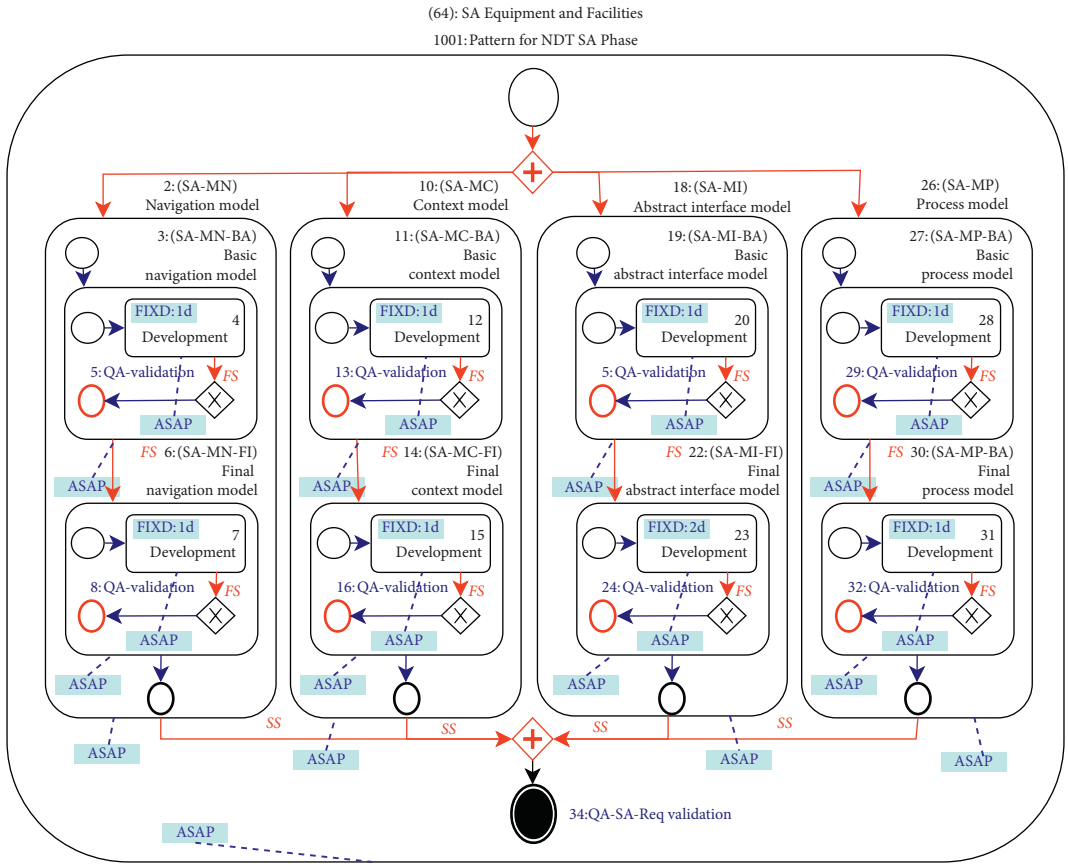


Figure 10: Generated BP: "Activity #64 SA of Facilities and Equipment." Control Flow of Activities and Time Rules (TCs and TDs) extracted from the project plan.

and *Processes* by means of *Time Dimension* identification as base heuristics.

We have compared reference NDT software business processes [31], manually designed by business experts, with processes generated by our MDA-based approach (such as processes corresponding to Figure 9 vs Figure 11 and Figure 10 vs Figure 12). It is worth mentioning that the generated processes are useful for business software experts and constitute a good approximation to those real that could be designed manually. Nevertheless, we can observe that the absence of advanced constructions (such as loops or transactions) in the generated processes is due to limitations of the source system (MPS®), which neither allows iterations over links nor more powerful logic rules for grouping tasks.

As the level of abstraction is concerned, we have been generating process instances from project plans, although sometimes *Activities* may be allocated to an *Activity* category in the scope of NDT reference [31] (cf. the use of patterns above). Therefore, our approach may generate process instances that are M_0 models regarding the Meta Object Facility (MOF) concept [8]. It may also capture a higher level of abstraction, such as M_1 models, in those cases where heuristics are applied to patterns or individual *Activities* are allocated to an *Activity* category. Process Mining Algorithms work in the same way, which means to know the task type of each executed, for instance, that is stored in the *event log*.

The approach could be easily applied to other contexts, such as (i) Project-Oriented Organizations, out of the business software sector, which also use MPS®; and (ii) New PMSs as sources, which are as widely used as MPS®, such as Redmine®. This way, the approach may be monetized running on a large number of projects carried out by many organizations.

Challenges to suite this approach to other PMSs are as simple as extracting a new PMS task Metamodel (as shown in Figure 4) and rewriting M2M mappings (Algorithm 1 and Table 4) from the new source by delivering a new roadmap (Figure 2). Business Processes may be enriched with other perspectives, such as *Organizational*, *Resources*, and *Costs*, which may be found in PMSs. Other Non-PAISs, which support the software lifecycle, could also play the role of sources (such as ECMs, ERPs, CRMs, SCMs, or Tailor-Made software). If multiple Non-PAISs were used in the same organization, then fragments of evidence regarding the same business process could be split into different databases. As a result, the approach would need to be strengthened in order to consistently merge process fragments into a unique conceptual process.

Generated processes with this approach are instances that we could consider raw results, which a human expert would have to review; however, we could consider the

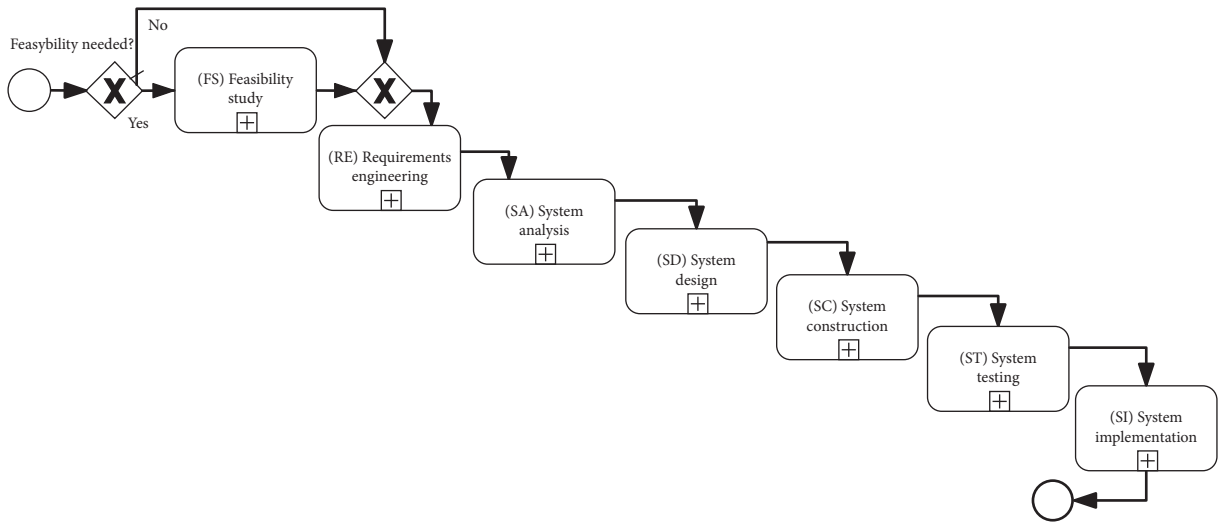


Figure 11: Navigational development technique (NDT). Waterfall software lifecycle.

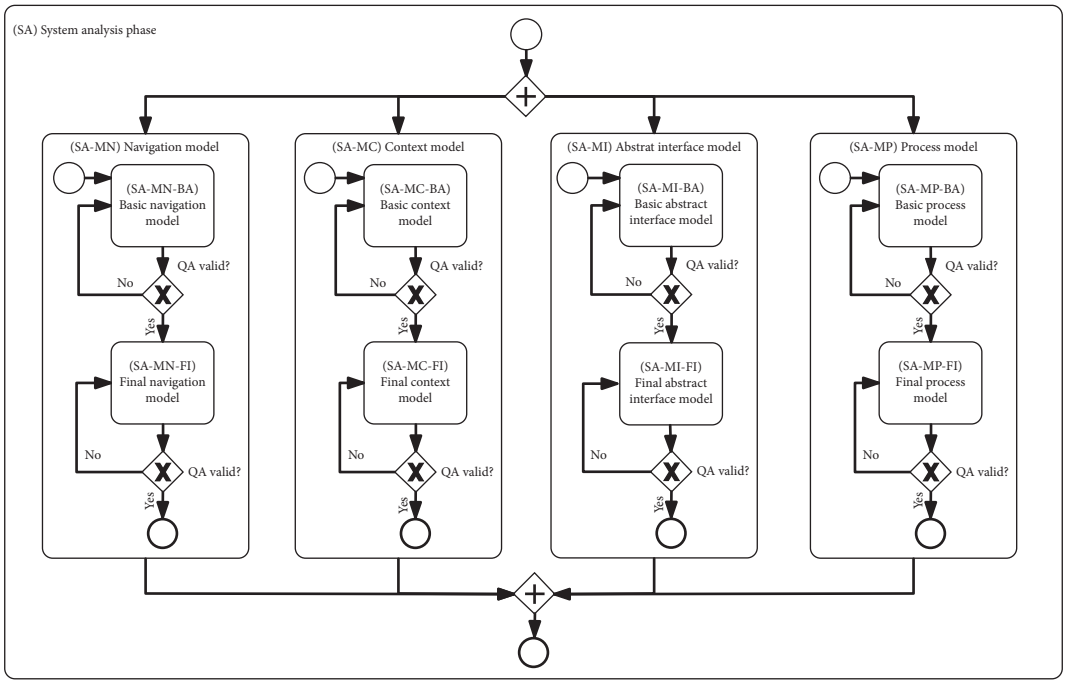


Figure 12: SA NDT phase manually modeled by an expert software engineer in NDT.

possibility of enriching the transformation algorithms with process refactoring patterns to improve the quality of them regarding parameters such as comprehensibility and modifiability [40–42].

6. Conclusion and Future Work

Software experts are increasingly using the BPM approach, making ITCs become more competitive in today's globalized world. Process Mining Techniques [17–19], as the major exponent of BPD, are well suited to obtain processes from PAIS *event logs*, but not from Non-PAISs because they lack these artifacts. Some Non-PAISs used by ITCs may hide a lot of knowledge about the execution of software business processes. Specifically, PMSs, such as MPS®, widely used by ITCs, is a source of project plans that may be transformed into processes, since they offer more advantages (i.e., BPM lifecycle of continuous improvement [2]) than just planning projects.

Regarding BPD from Non-PAISs, in which Process Mining is not applicable, we propose an MDA-based framework allowing different roadmaps to transform database artifacts, regarding tasks of ITCs, into processes of ITCs. Then, the proposal is useful to apply BPD to Non-PAISs of ITCs, such as PMSs. Based on this framework, we have developed a specific MDA-based roadmap (from *MPS® database* to *Process_Metamodel* [28]) to convert project plans, stored in an MPS® database, into software business

processes, which conform to *Process Metamodel* [28]. That allows interoperability with BPMLs (SPMLs or GPMLs) used in the software field. The challenges that arise with this approach are (i) to use LISs for software lifecycle management and (ii) to study and define the database metamodel of tasks that allows exploring the hidden dimensions (initially *Control flow* and *Time Dimension*, but extendable to others) of processes.

The related works we have further analyzed let extract processes from Non-PAISs by using heuristics or tools like SDG [36], ASTM [10, 23–27], and KDM [10, 23, 24]. Unlike our approach, they need a lot of extra effort to approximate to the reality of processes, so they have not been sufficiently applied to industrial contexts. To the best of our knowledge, we have not found out any study such as our MDA-based approach, whose main characteristics are as follows:

(i) It is initially focused on ITCs and PMSs as source systems since business processes are organized as projects whose states are stored in databases of PMSs.

(ii) Heuristics apply M2M transformations based on *Time Dimension* of source projects, which are in correspondence with processes defined with target *Process Metamodel* [28] (cf. Figure 1(a)); meanwhile, related approaches only use ASTM and KDM. Metamodeling-based reverse engineering procedure provides a high level of interoperability to our approach.

(iii) Results are processes close to business software expert, which may be defined with BPMLs. If the target BPML allows standard XML exchange formats, such as .xsd, .xmi, or .xpdl, then, results will be available to other roadmaps with the same effort. Later, the business expert must analyze and complete processes in a BPM lifecycle of continuous improvement [2].

To summarize, our proposal provides a framework to generate software business processes that would otherwise be hidden or wasted in databases of Non-PAISs. This hidden knowledge can be used to implement the BPM approach in ITCs that will help them to become more competitive and reduce costs. Compared to other BPD methods [10, 23–27, 36] used with Non-PAISs, our results are more adjusted to the reality of processes since we focus on transformations among artifacts that are close to executed processes that exist at different levels of abstraction (i.e., platform level and software expert level). Furthermore, business processes may be enriched with data regarding *Resources* and *Costs* that may also be bound to projects in PMSs. This way, new data will be available to set metrics and study Key Performance Indicators (KPIs) of software business processes.

This paper illustrates the AQUA-WS project case study [29] to test the developed MDA-based roadmap (i.e., from MPS® database to *Process_Metamodel* [28]). In this case study, we have shown that generated processes are similar to real processes that a business software expert may design. For this reason, we have delivered a semiautomatic proposal to obtain processes of ITCs.

We acknowledge that this study is only a first step towards validating the approach. With regard to future work,

(i) First, we plan to validate the approach with more cases following the steps:

  (a) Applying the proposal to more MPS® case studies with other software lifecycles, such as methodologies based on Agile Methods [66, 67]. The developed MPS® roadmap is reusable in many cases, thus the new inputs will be MPS® plans involving categorized activities concerning software development projects. Activities must be classified in a category in the same way that Process Mining Techniques [17–19] use traces stored in PAISs *event logs*, which means that it is necessary to know the type of each particular task.
  We are working in a prototype to reverse engineering MPS® databases of ITCs, which is based on Enterprise Architect (http://www.sparxsystems.com) customization. We are trying to organize the exploitation of this approach in the Andalusian government bodies (cf. the EMPOWER [68] project).

  (b) Testing the same approach either with other PMSs, such as Redmine®, with more types of Non-PAISs, such as ECMs (e.g., Alfresco® or SharePoint®), popular ERPs (licensed like SAP®, Oracle or Microsoft, or open sources like Open Bravo® or Odoo®) or Tailor-Made Software, among others.

  (c) Evaluating the approach out of the IT field, which means in other project-oriented industrial sectors that utilize PMSs.

  (d) Based on a set of significant cases, performing more solid statistical validation of the proposal to measure efficiency and effectiveness indicators of our approach for the automatic generation of software business processes in comparison with manual methods that an expert in this field could use.

(ii) Second, another line of research should cope with extending the *Process Metamodel* [28] (c.f. Figure 1(a)) to involve other process dimensions such as *Resource*, *Case*, and *Data*. As *Time Dimension* is concerned, the affinity of our Metamodel [28] with metamodels proposed by Awad et al. [69] and Stroppi et al. [70], related to *Resource* perspective, could enrich target process models including *Time* and *Resource* dimensions. In this case, the proposal will need to be extended to allow merging process fragments from different source legacy databases used by the same organization.

(iii) Third, generated processes are raw results that may lack a certain quality with respect to comprehensibility and modifiability. In this sense, we could

consider enriching the transformation algorithms by means of refactoring techniques [40–42] so that human-perceived quality measures could be improved.

(iv) Finally, we aim to generate standard *event logs*, such as XES format [19], for Non-PAISs. It should help to combine our approach with Process Mining Techniques [17–19] in order to compare proposed processes.

Data Availability

The AQUA-WS data used to support the findings of this study are available upon request to the authors.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Authors' Contributions

C. Arevalo provided the initial idea and guided the whole process of the manuscript. I. Ramos has contributed adapting meta-models of processes for the extraction of temporary rules from legacy systems. J. Gutiérrez has contributed to the creation of the general approach based on MDE for the generation of software processes from legacy systems used by ITCs. M. Cruz helped design the AQUA-WS case study, execute this specific roadmap to generate software processes, complete data analysis, and compare automatically generated processes with manual processes developed by ITC software experts.

Acknowledgments

This research has been supported by the POLOLAS project (TIN2016-76956-C3-2-R) of the Spanish Ministry of Economy and Competitiveness.

References

[1] W. M. van der Aalst, "Business process management: a comprehensive survey," *ISRN Software Engineering*, vol. 2013, Article ID 507984, 37 pages, 2013.

[2] M. Dumas, W. M. P. van der Aalst, and A. H. M. ter Hofstede, *Process-Aware Information Systems: Bridging People and Software through Process Technology*, John Wiley & Sons, Inc., Hoboken, NJ, USA, 2005.

[3] J. Luftman, H. S. Zadeh, B. Derksen, M. Santana, E. H. Rigoni, and Z. Huang, "Key information technology and management issues 2012-2013: an international study," *Journal of Information Technology*, vol. 28, no. 4, pp. 354–366, 2013.

[4] O. Henfridsson, L. Mathiassen, and F. Svahn, "Managing technological change in the digital age: the role of architectural frames," *Journal of Information Technology*, vol. 29, no. 1, pp. 27–43, 2014.

[5] S. Beecham, T. Hall, and A. Rainer, "Software process improvement problems in twelve software companies: an empirical analysis," *Empirical Software Engineering*, vol. 8, no. 1, pp. 7–42, 2003.

[6] F. Ruiz-González and G. Canfora, "Software process: characteristics, technology and environments, UPGrade," *The European Journal for the Informatics Professional*, vol. 5, no. 5, pp. 6–10, 2004, http://www.cepis.org/upgrade/files/full-2004-V.pdf.

[7] S. Kent, "Model driven engineering," in *Proceedings of the Third International Conference on Integrated Formal Methods*, pp. 286–298, Turku, Finland, May 2002.

[8] A. Kleppe, J. Warmer, and W. Bast, *MDA Explained: The Model Driven Architecture: Practice and Promise*, Addison-Wesley Professional, Boston, MA, USA, 2003, ISBN: 032119442X.

[9] P. Mohagheghi, W. Gilani, A. Stefanescu, and M. A. Fernandez, "An empirical study of the state of the practice and acceptance of model-driven engineering in four industrial cases," *Empirical Software Engineering*, vol. 18, no. 1, pp. 89–116, 2013.

[10] R. Pérez-Castillo, I. García-Rodríguez de Guzmán, M. Piattini, and Á. S. Places, "A case study on business process recovery using an e-government system," *Software: Practice and Experience*, vol. 42, no. 2, pp. 159–189, 2012.

[11] B. Gedik and H. Andrade, "A model-based framework for building extensible, high performance stream processing middleware and programming language for IBM InfoSphere Streams," *Software: Practice and Experience*, vol. 42, no. 11, pp. 1363–1391, 2012.

[12] X. Zhao and Y. Zou, "A business process-driven approach for generating software modules," *Software: Practice and Experience*, vol. 41, no. 10, 2011.

[13] J. Bisbal, D. Lawless, B. Wu, and J. Grimson, "Legacy Information Systems: Issues and Directions," *IEEE Software*, vol. 16, no. 5, pp. 103–111, 1999.

[14] W. M. Ulrich, *Legacy Systems: Transformation Strategies*, Prentice Hall, Englewood Cliffs, NJ, USA, 2002, ISBN 013044927X, http://dl.acm.org/citation.cfm?id=515375.

[15] R. C. Seacord, D. Plakosh, and G. A. Lewis, *Modernizing Legacy Systems: Software Technologies, Engineering Processes, and Business Practices*, Addison-Wesley, Boston, MA, USA, 2003, ISBN 0321118847, http://dl.acm.org/citation.cfm?id=599767.

[16] A. De Lucia, R. Francese, G. Scanniello, and G. Tortora, "Developing legacy system migration methods and tools for technology transfer," *Software: Practice and Experience*, vol. 38, no. 13, pp. 1333–1364, 2008.

[17] W. M. P. van der Aalst, *Process Mining: Discovery, Conformance and Enhancement of Business Processes*, Springer, Berlin, Germany, 2011.

[18] W. M. P. Van der Aalst, "A general divide and conquer approach for process mining," in *Proceedings of the Federated Conference on Computer Science and Information Systems (FedCSIS 2013)*, pp. 1–10, Kraków, Poland, September 2013, ISBN 9781467344715, http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6643968.

[19] H. M. W. Verbeek, J. C. A. M. Buijs, B. F. Van Dongen, and W. M. P. van der Aalst, "XES, XESame, and ProM 6," in *Proceedings of the International Conference on Advanced Information Systems Engineering (CAiSE Forum 2010)*, pp. 60–75, Hammamet, Tunisia, June 2011.

[20] W. M. P. van der Aalst, "Extracting event data from databases to unleash process mining," in *Management for Professionals*, pp. 105–128, Springer International Publishing, Berlin, Germany, 2015.

[21] S. Adam, N. Riegel, J. Doerr, O. Uenalan, and D. Kerkow, "From business processes to software services and vice versa-

an improved transition through service-oriented requirements engineering," *Journal of Software: Evolution and Process*, vol. 24, no. 3, pp. 237–258, 2012.

[22] Y. Zou, J. Guo, K. C. Foo, and M. Hung, "Recovering business processes from business applications," *Journal of Software Maintenance and Evolution: Research and Practice*, vol. 21, no. 5, pp. 315–348, 2009.

[23] R. Pérez-Castillo, M. Fernández-Ropero, I. G. R. de Guzmán, and M. Piattini, "MARBLE. A business process archeology tool," in *Proceedings of the 2011 27th IEEE International Conference on Software Maintenance (ICSM)*, pp. 578–581, IEEE, Williamsburg, VA, USA, September 2011.

[24] R. Pérez-Castillo, I. G.-R. de Guzmán, and M. Piattini, "Business process archeology using MARBLE," *Information and Software Technology*, vol. 53, no. 10, pp. 1023–1044, 2011.

[25] C. Arevalo, *Una prop uesta basada en el paradigma dirigido por modelos para la extracción de procesos del software desde sistemas heredados utilizando la perspectiva temporal*, Ph.D. thesis, University of Seville, Seville, Spain, 2016, https://idus.us.es/xmlui/handle/11441/42996.

[26] C. Arevalo, M. T. Gómez-López, A. M. Reina Quintero, and I. Ramos, "An architecture to infer business rules from event condition action rules implemented in the persistence layer," in *Uncovering Essential Software Artifacts through Business Process Archeology*, R. Perez-Castillo and M. Piattini, Eds., Business Science Reference, Hershey, PA, USA, pp. 201–221, 2013.

[27] C. Arevalo, I. Ramos, and M. J. Escalona, "Discovering business models for software process management—an approach for integrating time and resource perspectives from legacy information systems," in *Proceedings of the 17th International Conference on Enterprise Information Systems (ICEIS 2015)*, pp. 353–359, SCITEPRESS - Science and Technology Publications, Barcelona, Spain, April 2015.

[28] C. Arevalo, M. J. Escalona, I. Ramos, and M. Domínguez-Muñoz, "A metamodel to integrate business processes time perspective in BPMN 2.0," *Information and Software Technology*, vol. 77, pp. 17–33, 2016.

[29] C. R. Cutilla, J. A. García-García, J. J. Gutiérrez et al., "Model-driven test engineering: a practical analysis in the AQUA-WS project," in *Proceedings of the 7th International Conference on Software Paradigm Trends (ICSOFT 2012)*, pp. 111–119, Rome, Italy, July 2012, ISBN:9789898565198, http://www.scopus.com/inward/record.url?eid=2-s2.0-84868673216&partnerID=tZOtx3y1.

[30] J. Ponce, L. García-Borgoñon, J. A. García-García et al., "A model-driven approach for business process management," *Covenant Journal of Engineering & Technology (CJICT)*, vol. 1, no. 2, pp. 32–52, 2013, http://journals.covenantuniversity.edu.ng/index.php/cjict/article/view/79.

[31] M. J. Escalona, M. Mejías, J. Torres, and A. Reina, "The NDT development process," in *Proceedings of the International Conference on Web Engineering*, pp. 463–467, Springer, Oviedo, Spain, July 2003.

[32] P. Newcomb, "Architecture-driven modernization (ADM)," in *Proceedings of the 12th Working Conference on Reverse Engineering (WCRE'05)*, p. 237, IEEE, Pittsburgh, PA, USA, November 2005.

[33] R. Pérez-Castillo, I. García-Rodríguez de Guzmán, I. Caballero, and M. Piattini, "Software modernization by recovering web services from legacy databases," *Journal of Software: Evolution and Process*, vol. 25, no. 5, pp. 507–533, 2013.

[34] E. J. Chikofsky and J. H. Cross, "Reverse engineering and design recovery: a Taxonomy," *IEEE Software*, vol. 7, no. 1, pp. 13–17, 1990.

[35] E. J. Byrne, "Software reverse engineering: a case study," *Software: Practice and Experience*, vol. 21, no. 12, pp. 1349–1364, 1991.

[36] A. Cleve, J. Henrard, and J. Hainaut, "Data reverse engineering using system dependency graphs," in *Proceedings of the 2006 13th Working Conference on Reverse Engineering*, pp. 157–166, IEEE, Benevento, Italy, October 2006.

[37] OMG, "Unified modeling language (UML®)," 2015, http://www.omg.org/spec/UML/.

[38] V. Cosentino and S. Martinez, "Extracting UML/OCL integrity constraints and derived types from relational databases," in *Proceedings of the 13th International Workshop on OCL, Model Constraint and Query Languages*, pp. 43–52, Miami, FL, USA, September 2013, http://hal.univ-nantes.fr/hal-00869231/.

[39] M. Zanoni, F. Perin, F. A. Fontana, and G. Viscusi, "Pattern detection for conceptual schema recovery in data-intensive systems," *Journal of Software: Evolution and Process*, vol. 26, no. 12, pp. 1172–1192, 2014.

[40] D. Caivano, M. Fernández-Ropero, R. Pérez-Castillo, M. Piattini, and M. Scalera, "Artifact-based vs. human-perceived understandability and modifiability of refactored business processes: an experiment," *Journal of Systems and Software*, vol. 144, pp. 143–164, 2018.

[41] R. Pérez-Castillo, M. Fernández-Ropero, M. Piattini, and D. Caivano, "How does refactoring affect understandability of business process models?," in *Proceedings of the 25th International Conference on Software Engineering & Knowledge Engineering (SEKE'13)*, pp. 644–649, Knowledge Systems Institute Graduate School, Boston, MA, USA, June 2013, ISBN-13: 978-1-891706-33-2.

[42] R. Pérez-Castillo, M. Fernández-Ropero, and M. Piattini, "Business process model refactoring applying IBUPROFEN. An industrial evaluation," *Journal of Systems and Software*, vol. 147, pp. 86–103, 2019.

[43] L. García-Borgoñón, M. A. Barcelona, J. A. García-García, M. Alba, and M. J. Escalona, "Software process modeling languages: a systematic literature review," *Information and Software Technology*, vol. 56, no. 2, pp. 103–116, 2014.

[44] C. Portela, A. Vasconcelos, A. Silva et al., "A comparative analysis between BPMN and SPEM modeling standards in the software processes context," *Journal of Software Engineering and Applications*, vol. 5, no. 5, pp. 330–339, 2012.

[45] R. M. Pillat, T. C. Oliveira, P. S. C. Alencar, and D. D. Cowan, "BPMNt: a BPMN extension for specifying software process tailoring," *Information and Software Technology*, vol. 57, pp. 95–115, 2015.

[46] ISO, *ISO/IEC 19510: OMG Business Process Model and Notation (BPMN)*, International Organization for Standardization, Geneva, Switzerland, 2013.

[47] OMG, "SPEM, software & systems process engineering metamodel specification 2.0," 2008, http://www.omg.org/spec/SPEM/.

[48] R. Bendraou, C. Benoît, X. Xavier, and M.-P. Gervais, "Definition of an executable SPEM 2.0," in *Proceedings of the Asia-Pacific Software Engineering Conference, APSEC*, pp. 390–397, Nagoya, Japan, December 2007.

[49] I. Ruiz-Rube, J. M. Dodero, M. Palomo-Duarte, M. Ruiz, and D. Gawn, "Uses and applications of software & systems process engineering meta-model process models. A

systematic mapping study," *Journal of Software: Evolution and Process*, vol. 25, no. 9, pp. 999–1025, 2013.

[50] ISO, *ISO/IEC 24744: Software Engineering—Metamodel for Development Methodologies*, International Organization for Standardization, Geneva, Switzerland, 2014.

[51] OMG, "Essence—kernel and language for software engineering methods (essence)," 2015, http://www.omg.org/spec/Essence/.

[52] I. Jacobson, P.-W. Ng, P. E. McMahon, I. Spence, and S. Lidman, "The essence of software engineering: the SEMAT kernel," *Communications of the ACM*, vol. 55, no. 12, pp. 42–49, 2012.

[53] F. Bonnet, G. Decker, L. Dugan, M. Kurz, and Z. Misiak, "Making BPMN a true lingua franca, BPM Trends," 2014, http://www.bptrends.com/bpt/wp-content/uploads/06-03-2014-ART-MakingBPM-a-TrueLinguaFranca-Zbigniew-Misiak-etal.pdf.

[54] C. Flores and M. Sepúlveda, "Temporal specification of business processes through project planning tools," in *Business Process Management Workshops*, pp. 85–96, Springer Berlin Heidelberg, Berlin, Germany, 2011.

[55] D. Gagné and A. Trudel, "Time-BPMN," in *Proceedings of the IEEE Conference on Commerce and Enterprise Computing*, pp. 361–367, Vienna, Austria, July 2009.

[56] S. Cheikhrouhou, S. Kallel, N. Guermouche, and M. Jmaiel, "Toward a time-centric modeling of business processes in BPMN 2.0," in *Proceedings of International Conference on Information Integration and Web-Based Applications & Services (IIWAS'13), Proceedings of the ACM International Conference Proceeding Series*, pp. 154–163, Vienna, Austria, 2013.

[57] E. González López De Murillas, W. M. P. van der Aalst, and H. A. Reijers, "Process mining on databases. Unearthing historical data from redo logs," in *Proceedings of the 3th International Conference on Business Process Management*, pp. 367–385, Innsbruck, Austria, August-September 2015.

[58] E. González López de Murillas, H. A. Reijers, and W. M. P. van der Aalst, "Connecting databases with process mining: a meta model and toolset," in *Proceedings of the International Conference on Business Process Modeling, Development and Support (BPMDS 2016)*, pp. 231–249, Ljubljana, Slovenia, June 2016.

[59] S. Jablonski and C. Bussler, *Workflow Management: Modeling Concepts, Architecture and Implementation*, International Thomson Computer Press, Boston, MA, USA, 1996, http://www.citeulike.org/group/6987/article/3401448.

[60] R. G. Ross, *Business Rules Concepts: Getting to the Point of Knowledge*, Business Rule Solutions, LLC, Houston, TX, USA, 4th edition, 2009, https://www.brsolutions.com/b_concepts.php.

[61] G. Wagner, "Rule modeling and markup," in *Reasoning Web*, pp. 251–274, Springer, Berlin, Germany, 2005.

[62] R. Engel, W. Krathu, M. Pichler et al., "Analyzing inter-organizational business processes," *Information Systems and e-Business Management*, vol. 14, no. 3, pp. 577–612, 2016.

[63] C. Gonzalez-Perez and B. Henderson-Sellers, *Metamodelling for Software Engineering*, Wiley Publishing, Hoboken, NJ, USA, 2008, ISBN 0470030364, 9780470030363, http://dl.acm.org/citation.cfm?id=1502365.

[64] J. F. Allen, "Maintaining knowledge about temporal intervals," *Communications of the ACM*, vol. 26, no. 11, pp. 832–843, 1983.

[65] OMG, "QVT, Meta Object Facility (MOF) 2.0 query/view/transformation," 2011, http://www.omg.org/spec/QVT/1.1/.

[66] K. Schwaber, "Scrum development process, business object design and implementation," 1997, http://link.springer.com/10.1007/978-1-4471-0947-1_11.

[67] J. Bosch and P. M. Bosch-Sijtsema, "Introducing agile customer-centered development in a legacy software product line," *Software: Practice and Experience*, vol. 41, no. 8, pp. 871–882, 2011.

[68] J. A. Garcia-Garcia, J. G. Enriquez, L. Garcia-Borgonon, C. Arevalo, and E. Morillo, "A MDE-based framework to improve the process management: the EMPOWER project," in *Proceedings of the 2017 IEEE 15th International Conference on Industrial Informatics (INDIN)*, pp. 553–558, IEEE, Emden, Germany, July 2017.

[69] A. Awad, A. Grosskopf, A. Meyer, and M. Weske, *Enabling Resource Assignment Constraints in BPMN*, Hasso Plattner Institute, Potsdam, Germany, 2009, http://oryx-project.org/pub/Public/AhmedAwad/Enabling_Resource_Assignment_Constraints_in.pdf.

[70] L. Stroppi, O. Chiotti, and P. Villarreal, *Extended Resource Perspective Support for BPMN and BPEL*, CIbSE, London, UK, 2012.

Hindawi
Scientific Programming
Volume 2019, Article ID 2074329, 9 pages
https://doi.org/10.1155/2019/2074329

Research Article

Study of Urban System Spatial Interaction Based on Microblog Data: A Case of Huaihe River Basin, China

Yong Fan,[1,2] **Juhui Yao**,[3] **Zongyi He**,[4] **Biao He**,[1] **and Minmin Li**[1]

[1]*Research Institute for Smart Cities, School of Architecture and Urban Planning, Shenzhen University, Shenzhen, China*
[2]*College of Geographic Sciences, Xinyang Normal University, Xinyang, China*
[3]*Navinfo (Xi'an) Co. Ltd., Xi'an, China*
[4]*School of Resource and Environmental Sciences, Wuhan University, Wuhan, China*

Correspondence should be addressed to Biao He; whu_hebiao@hotmail.com

Received 6 November 2018; Accepted 17 January 2019; Published 5 February 2019

Guest Editor: Alvaro Rubio-Largo

The spatial interaction of urban system has been an hot research issue in the field of urban research. In this paper, user's microblog spatial information data were used to discern the spatial structure of an urban area. Firstly, Sina Weibo microblog data for 2011–2015 were used to establish a thematic database of cities along the Huaihe River Basin, China. Secondly, network connectivity, inflow, and outflow of three indicator systems were analyzed. Finally, combining this database with socioeconomic data, experimental verification and comparative analysis were carried out. The study found that the urban spatial relation in the Huaihe River Basin has the following characteristics: the spatial difference of urban size distribution is obvious; urban layout presents a stratified aggregation phenomenon; and the high-grade cities lead the city's interaction. The research shows that this method of data mining for urban interaction in the Huaihe River Basin is valid and that this research into urban spatial patterns of river basins is applicable to other areas.

1. Introduction

In the four decades since China's reform, the urban population has increased from 170 million in 1978 to 790 million in 2016 and the urbanization rate has increased from 17.9% to 57.4% [1]. However, the level of urbanization in China is still lower than that of 70%–80% of developed countries. In the future, China will still be in the stage of rapid urbanization and development. The relationship between man and nature has become a hot topic in China at the same time. Urbanization promotes the exchange of materials, energy, personnel, and information among cities; this exchange is called urban space interaction [2]. How to effectively model and quantitatively express the spatial interaction of regional cities is a significant question and the basis of this research.

The regional urban system is a complex network [3, 4]. The connection between two nodes in this network can be represented by visible lines, such as highways and railways, or by invisible lines, such as aviation pathways, navigation, and the Internet [5]. The flow within the system emphasizes the value of urban nodes in shaping the whole network system, which provides a theoretical framework and an important starting point for the study of urban networks [6]. Data relate to traffic, and information flow can most directly reflect the degree of urban spatial interaction [7]. Early research on this aspect studied the transmission of mail. The scope of research on transportation is more comprehensive, including the highway, railway, and aviation networks [8, 9]. In addition, the urban hierarchy can also be explored through the distribution of corporate headquarters [10] and banks [11]. The flow and consumption of network information can reflect the level of urbanization [12], which can provide basic data for geographical and sociological studies [13, 14].

The age of big data and the rapid development of LBS (location-based service) allow for urban research to be conducted by novel methods [15, 16]. With the gradual

penetration of computers into various professions, social media data [17–19] have attracted the attention of geographers due to its use in research [20]. The use of social media was one of the needs of urban communities [21]. As for the typical spatial-temporal characteristics of social media data, many data mining research papers have analyzed information, such as urban hot events [22], identifying urban Corridors [23], geography of happiness [24], whereabouts of people [25], urban function connectivity [26], urban spatial interaction [27], revaluating urban space [28], and characterizing witness accounts [29]. Data mining [30] provides new tools, methods, and ideas for new geographical research, improving the efficiency, accuracy, and precision of the analysis [31].

Based on the data of Sina Weibo microblog [32], this paper established a thematic database of cities along the Huaihe River Basin, China. The spatial interaction of the regional urban system was analyzed from temporal and spatial characteristics of the data. Finally, combining this database with socioeconomic data, experimental verification and comparative analysis were carried out. This paper provides a reference for the exploration of the spatial relationship of regional urban systems and the driving mechanism for spatial structure of urban system.

2. Study Area

The Huaihe River Basin is located in the eastern region of China, between the Yangtze River Basin and the Yellow River Basin. It flows through the four provinces of Henan, Anhui, Jiangsu, and Shandong and covers a total area of $2.7 \times 10^5\,\mathrm{km}^2$. The Huaihe River Basin has multiple transitional nature and society and connects coastal region and inland region, as well as being the economic integration hub of the Yangtze River and Yellow River economic belts.

The Huaihe River Basin has an enormous population of 170 million people, accounting for 12.3 percent of the total population of China. The average population density is 611 people per km^2, which is 4.8 times of the average level across the country. Given that the Huaihe River Basin is a natural geographical unit and does not coincide with the boundaries of the administrative units, the study only selected the 26 cities whose governmental units are entirely in the Huaihe River Basin for calculation and analysis (Figure 1).

3. Data Source and Processing

3.1. Sina Weibo Microblog. The data gathered from the Sina Weibo microblog are the information released by microblog users on the social network. Compared with traditional social survey data and economic social statistics, the Sina Weibo microblog data have an unparalleled advantage in the fact that it performs well in real-time and is reliable and diverse. Currently, information can be gathered from the Sina Weibo microblog using web crawlers and the Sina Weibo microblog open platform API. This study uses the latter to obtain the relevant information for this research.

3.2. Preprocessing the Sina Weibo Microblog. First, the gathered data were cropped to the latitude and longitude range of the survey region. Each city administrative unit was divided into a 10 km-long grid, the coordinates of the grid center were obtained, and the WGS-84 geographic coordinate system was applied. Next, taking the longitude and latitude of different cities as parameters, user ID information from these cities was obtained through the nearby_timeline interface (an API interface). The user ID information includes a source (through the authorized APPKey), longitude and latitude, as well as other data. Using the user ID information as the parameter the user's historical data through the check-in interface (another API interface) were obtained. The data returned by calling the API interface were in the JSON format. In this study, a Python script was written to achieve batch data acquisition. The process of data acquisition is shown in Figure 2.

3.3. The Spatial Interaction of the Urban System. Microblog check-in data can reflect the spatial interaction of urban system; this paper adopts two methods to analyze these interactions. The first is connectivity, which is used to analyze the intensity of communication between regional cities; the second is flow, through which the user's trajectory and the interactive state among cities can be analyzed.

3.3.1. Connectivity. This method is based on the city's check-in data, from which the network connectivity among cities was analyzed, which is described as follows:

(1) *Variable Unitization.* After variable unitization, a matching matrix was established regarding the city check-in data using the following equation:

$$V_{ij}0 = \frac{V_{ij}1}{\sum_{j=1}^{25} V_{ij}}, \tag{1}$$

where $V_{ij}0$ is a microblog user in city i who signed into city j after unitization; $V_{ij}1$ represents a microblog user of i city who signed in city j after data acquisition; and $\sum V_{ij}$ represents the sum of the check-in data of user form city i in city j.

(2) *Urban External Connection Index.* The urban external connection index represents the difference between the total number of check-ins in other cities and those in city i. This is calculated using the following equation:

$$N_i0 = \sum_{j=1}^{25} V_{ij}0 - V_{ii}0, \tag{2}$$

where N_i0 is the external connection index of city i, which reflects the check-in data (V) of city i in the other 25 cities; $V_{ii}0$ represents the check-in data of city i after unitization; and $V_{ij}0$ is the sum of check-in data of microblog users from city i who signed in other cities after unitization.

Figure 1: Study area.

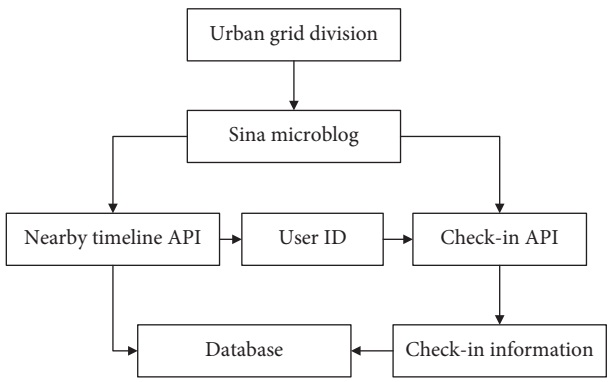


Figure 2: The process of sina Weibo microblog data acquisition.

If $N_i0 > 0$, users of city i sign into the microblog more when not in city i; if $N_i0 < 0$, users registered to city i check into Sina Weibo locally. $N_i0 = 0$ means that users from city i are equally likely to sign into the microblog whether they are within city i or away from the city.

(3) *Urban Network Connectivity*. First, the standardized check-in data of city i in city j and the standardized check-in data of city j in city i are multiplied together to obtain the urban network connectivity. Then, these data are standardized to produce the standardized network connectivity. This is summarized in the following equation:

$$R_{ij} = V_{ij}0 * V_{ji}0,$$
$$R_{ij}0 = R_{ij} * \left(\frac{100}{\mathrm{Max}\left(R_{ij}\right)}\right), \quad (3)$$

where $R_{ij}0$ is the standardized network connectivity between city i and city j; R_{ij} is the network connectivity between city i and city j; $V_{ij}0$ is the standardized value of friends of city j residents who live in city i; $\mathrm{Max}(R_{ij})$ is the maximum network connectivity value; and $R_{ij}0$ reflects interconnectedness of each city's information.

(4) *Network Connectivity in Each City*. M_i0 is the standard network connectivity of city i with itself.

M_i0i is the difference between the standard network connectivity of city i and city j and the standard network connectivity of city i itself. This is calculated using the following equation:

$$M_i0 = \sum_{j=1}^{25} R_{ij}0 - R_{ii}0, \quad (4)$$

where M_i0 reflects the linkage strength of city i within the network system.

3.3.2. Inflow Rating and Outflow Rating. The inflow and outflow ratings reflect the attractiveness and interaction of cities, respectively. The inflow rating is the total number of other cities signed into the city. The greater the inflow, the higher the attractiveness and degree of interaction the city has. The outflow rating reflects the total number of users registered to the city signed into other cities.

The greater the outflow rating, the greater the output of the city, which indicates the city has a higher degree of interaction with other cities.

Formula (5) is used to calculate the inflow rating of city i, where V_{i1} indicates the number of check-ins from users registered to city j in city i:

$$V_{i1} = \sum_{j=1}^{25} V_{ij}. \quad (5)$$

Formula (6) is used to calculate the outflow rating of city i, where V_{i2} indicates the number of check-ins from users registered to city j in city i:

$$V_{i2} = \sum_{j=1}^{25} V'_{ij}. \quad (6)$$

To further analyze the reasons for flow among cities, the ratio of inflow to outflow is introduced, which can reflect the population flow among cities. This is calculated using the following equation:

$$R = \frac{V_{i1}}{V_{i2}}. \quad (7)$$

4. Results

4.1. Connectivity Analysis. Based on the calculation of urban connectivity, the microblog check-in data in the study area can be calculated. From this, the urban external connectivity index and network connectivity are obtained. This data are summarized in Table 1.

In Table 1, the N_i0 value of Luohe and Kaifeng is 1, which shows that there are more check-ins from outside users into both Luohe and Kaifeng. The N_i0 value of Rizhao is −0.60, which is the lowest in the Huaihe River Basin. This indicates that check-in data of Rizhao consist of nearly only local users. In addition, the M_i0 value of Bengbu is −1543.59, which is the minimum in the Huaihe River Basin. Bengbu's external relations with the basin as a whole are relatively weak. The M_i0 of Luohe is 291.15, which is the largest in the Huaihe River Basin. It shows that Luohe has a strong external relationship with the whole basin. Luohe is located in the middle of the Henan Province, next to the provincial capital Zhengzhou. The city has convenient transportation, prosperous economy, well-developed tourism industry, and large population mobility; these factors allow for more users to travel to other locations and increase the M_i0 value of Luohe.

Table 1: External connection index and network connectivity.

Name	N_i0	M_i0
Luohe	1.00	291.15
Kaifeng	1.00	203.67
Huainan	0.97	159.24
Bozhou	0.95	120.38
Yangzhou	0.86	−84.64
Fuyang	0.62	−47.69
Xinyang	0.42	−83.31
Lianyungang	0.39	−75.32
Jining	0.32	−74.07
Suzhou	0.23	−86.43
Huaiyin	−0.01	−96.93
Zhumadian	−0.08	−97.96
Suqian	−0.15	−96.90
Heze	−0.16	−98.96
Zhoukou	−0.22	−98.30
Bengbu	−0.25	−1543.59
Xuchang	−0.30	−96.77
Liuan	−0.31	−98.11
Zaozhuang	−0.31	−98.63
Pingdingshan	−0.36	−97.27
Huaibei	−0.50	−95.69
Shangqiu	−0.51	−97.60
Linyi	−0.53	−97.88
Xuzhou	−0.53	−96.98
Yancheng	−0.59	−90.80
Rizhao	−0.60	−99.02

By analyzing the relationship between the external and internal connectivity of a city (Figure 3), the following three conclusions were reached:

(1) The internal urban network connectivity is consistent with the urban external connectivity: the lower the city ranks in terms of internal connectivity, the lower the city's level of external connections with the rest of the network. Among the 26 cities, Luohe, Kaifeng, Huainan, and Bozhou comprised the top four. The positive values for these cities indicate that the intensity of external connections between the four cities is greater than that within the cities. The remaining 22 cities have negative values, which show that these cities have low external connections and that communication within these cities is greater than that with the external network. The network connectivity of Fuyang and Bengbu does not correspond to the urban city hierarchy which shows that, although the interconnectedness of these 2 cities with the other cities in the basin is not high, their absolute degree of network connectivity is relatively high. This is mainly due to the strong network connectivity in a small area. This phenomenon is most obvious in Bengbu.

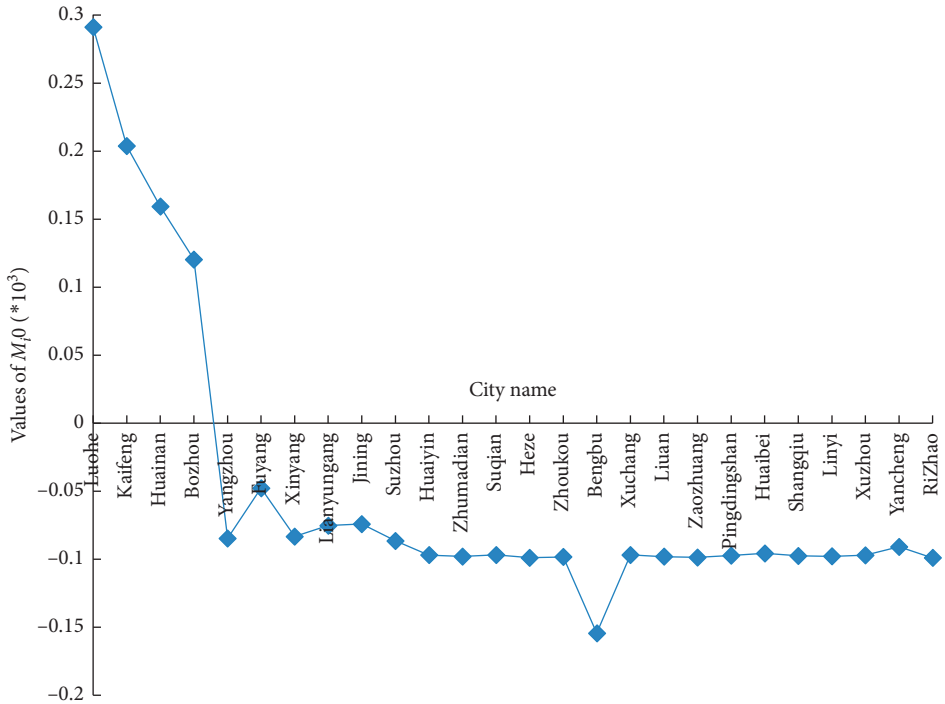


Figure 3: Relationship between urban network connectivity and urban hierarchy.

(2) Foreign contact intensity is related to a city's economic level: the standardized network connectivity (M_i0) reflects the intensity of a city's linkage to the entire basin system. M_i0 is derived from the difference between the standardized network connectivity value of city *i* with the other cities and the value of the city *i* within itself. M_i0 reflects the strength of the external connection of city *i*. Based on the classification of natural breakpoints (Jenks classification), the 26 cities are divided into 5 grades, which are shown in Table 2.

According to the statistical results of the four provinces GDP (Jiangsu, Shandong, Anhui, and Henan), the 26 cities are ranked in the order of Jiangsu, Shandong, Henan, and Anhui. The five cities of Luohe, Kaifeng, Huainan, Bozhou, and Fuyang in Henan and Anhui were ranked the highest. This is because the population of the two cities generally interacts on a local scale.

(3) Significant difference in connection intensity between cities: the network connectivity of 26 cities was sorted were obtained. From the data, it was found that the strength of the city-to-city relationship varies greatly. There are 225 sets of data between 0 and 1, occupying 69.23% of the total, and 62 sets of data between 1 and 10, occupying 19.08% of the total. Moreover, 31 sets of data between 10 and 100 accounted for 9.54% of the data, while 16 sets of 100–1000 accounting for 4.92%.

4.2. Urban Interaction Analysis. According to the 26 cities' check-in data and the user's mobile track, the urban interaction of the 26 cities can be analyzed.

Table 2: Classification of foreign contact intensity.

Grade	City	Connectedness value
1	Luohe, Kaifeng, Huainan, Bozhou	−47.69 to 291.45
2	Fuyang	−74.07 to −47.69
3	Jining, Lianyungang, Xinyang, Yangzhou, Suzhou	−90.80 to −74.07
4	RiZhao, Heze, Zaozhuang, Zhoukou, Liuan, Zhumadian, Linyi, Shangqiu, Pingdingshan, Xuzhou, Huaiyin, Suqian Xuchang, Huaibei, Yancheng, Suzhou, Yangzhou, Xinyang	−154.36 to −90.80
5	Bengbu	−154.36

4.2.1. Classification of Inflow Ratings. The inflow rating of the city reflects the total number of microblog entries from other cities signed in the city. Taking city *i* as the research object, the check-in data of city *j* in city *i* are collected, and then the total inflow of city *i* is obtained. According to the natural breakpoint, the 26 cities are divided into 5 grades, as shown in Table 3.

From Table 3, we can see that the inflow ratings of the cities have the following characteristics:

(1) The difference in inflow level is large: the largest number of check-ins to a city is 96,000, while the smallest city had less than 10,000 check-ins.

(2) The fourth city rank had the most of cities; most cities had between 10,000 and 20,000 check-ins.

(3) The check-in data are unevenly distributed: The data are mainly distributed in the northeast, southeast, and east of the basin, mainly in the Jiangsu and Shandong provinces. The amount of data in the Anhui and Henan provinces is relatively small.

Table 3: Classification of inflow rating.

Grade	City	Inflow rating
1	Jining, Xuzhou, Bengbu, Yancheng	60,001–96,000
2	Shangqiu, Huaibei, Huaiyin	40,001–60,000
3	Kaifeng, Linyi, Lianyungang, Yangzhou, Liuan	20,001–40,000
4	Pingdingshan, Xuchang, Luohe, Xinyang, Zhoukou, Fuyang, Bozhou, Huainan, Suzhou, Zaozhuang, Suqian, Rizhao	10,001–20,000
5	Heze, Zhumadian	10,000

Table 4: Classification of outflow rating.

Grade	City	Outflow rating
1	Yangzhou	80,001–102,087
2	Xinyang, Bozhou, Jining, Lianyungang	50,001–80,000
3	Luohe, Kaifeng, Fuyang, Linyi, Huaiyin	30,001–50,000
4	Pingdingshan, Xuchang, Zhoukou, Zhumadian, Shangqiu, Heze, Suzhou, Xuzhou, Zaozhuang, Suqian, Huainan, Bengbu, Luan, Yancheng	10,001–30,000
5	Huaibei, Rizhao	10,000

4.2.2. Classification of Outflow Ratings. The outflow rating reflects the total number of users from a city that signed into other cities. Similarly to the inflow rating, the data were classified according to the natural breakpoints, and the results shown in Table 4 were obtained.

From Table 4, it can be seen that the outflow has the following characteristics.

(1) The difference in outflow level varies significantly across the basin: the largest number of check-ins from one city is more than 100,000, while the smallest city has less than 10,000.

(2) The fourth city rank is the largest in terms of outflow: most cities have between 10,000 and 30,000 check-ins in other cities.

(3) The spatial distribution of the outflow data is uneven: the data are mainly distributed in the central and western regions of the basin. Out of all the provinces, the outflow from the Jiangsu province is the largest, Shandong province is unevenly distributed, and Henan and Anhui provinces have more homogeneous outflow spatial distributions.

4.2.3. Ratio of Inflow and Outflow. The ratio of inflow and outflow can directly reflect the flow direction of population between cities and can also help to further analyze the driving factors of population flow between cities. The ratio of inflow and outflow can reflect the attractiveness of a city. The higher the ratio, the greater the attractiveness of the recipient city, or less attractive the city from which the people are migrating. According to natural breakpoint classification, the 26 cities are divided into 5 grades (Table 5).

From Table 5, the following ratio of inflow and outflow characteristics can be determined:

(1) The difference between the ratios of inflow and outflow between cities is large: the largest ratio is 1244 times more than the smallest.

(2) Most of the cities had inflow/outflow ratios below 1, and these cities were distributed more evenly. This shows that the inflow of most cities is less than that of the outflow.

(3) The ratio of inflow and outflow in Luohe, Kaifeng, and Huainan are relatively large, which indicates that the inflow and outflow data of these three cities are significantly different from those of the other cities in the river basin.

Table 5: Ratio of inflow and outflow.

Grade	City	Inflow and outflow ratio
1	Luohe	>300
2	Kaifeng, Huainan	51–300
3	Bozhou, Yangzhou	6–50
4	Xinyang, Fuyang, Suzhou, Jining, Lianyungang Pingdingshan, Xuchang, Zhoukou, Zhumadian, Shangqiu	1–5
5	Heze, Huaibei, Bengbu, Lu'an, Huaiyin, Suqian, Xuzhou, Zaozhuang, Linyi, Rizhao, Yancheng	<1

5. Discussion

In this paper, we used microblog data to effectively model and quantitatively express the spatial interaction of regional cities. Microblog data can not only reflect mobile information of users' tracks but also reflect users' travel and activity rules. Does socioeconomic status affect users' travel and activity rules? How does it affect? In order to answer these questions, this paper analyzes the correlation among urban microblog registration data, urban population, and GDP with the statistical data.

5.1. The Relationship between Urban Interaction and Social Economy. Take the GDP and the urban population as the abscissa, respectively, and the total city check-in data as the ordinates, Figures 4 and 5 are obtained.

From Figures 4 and 5, it can be concluded that there is no positive correlation among the total amount of urban check-in, the economic links in the entity and the total population. To a certain extent, this reflects the degree of urban interaction is a mechanism of the joint action of the economic level and population factors of the city. The city's largest inflow and local check-in data are the side portraits of the city's attractiveness index, which determines the level of city interaction. Cities with high interactive levels, such as Xuzhou and Yancheng, have high inflows and local check-in volumes, and their GDP is also higher in the provinces. Suqian, Suzhou, Zhoukou, Pingdingshan, Heze, Zhumadian, and other cities are based on the local sign of data, so their inflow is less and the interaction between cities is not strong. From the data of the Statistical Yearbook of 2015, it is not difficult to find that the cities with strong interactive ranks have higher GDP levels. Therefore,

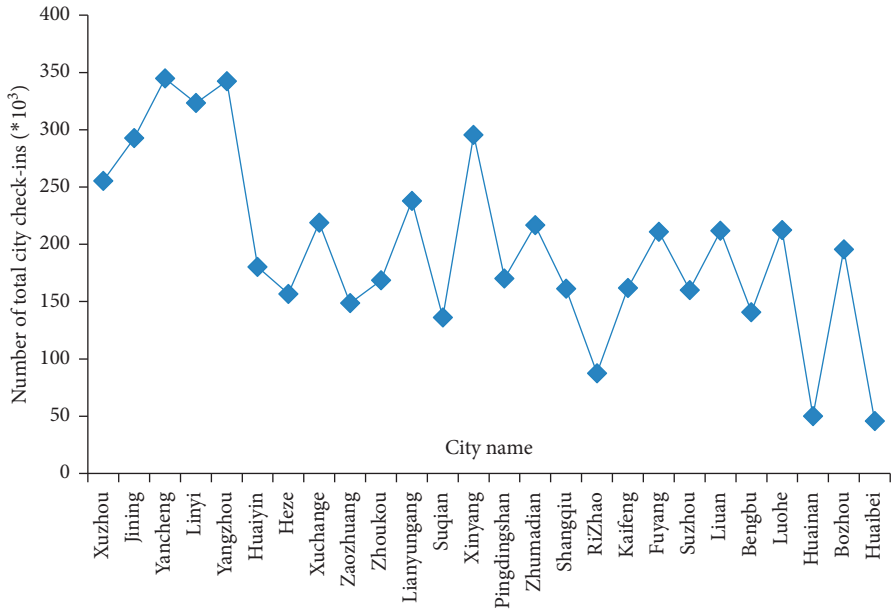


Figure 4: Total attendance ordered by GDP.

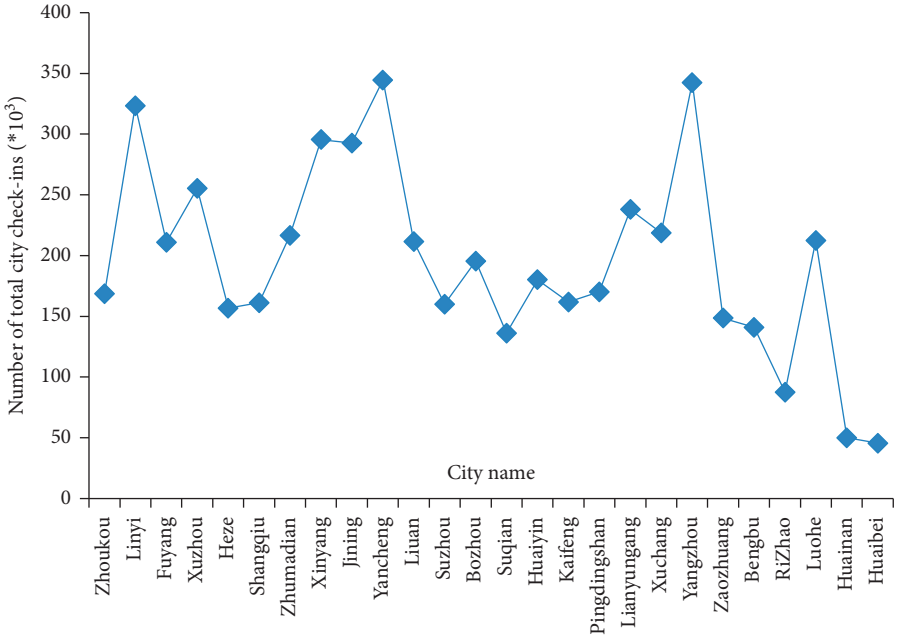


Figure 5: Total attendance ordered by population.

if we want to improve the city's GDP, we must strengthen the interaction between cities.

5.2. Microblog Data and Traditional Data. As a class of big data, microblog data contain user's location information which is different from traditional data, such as highway data and railway data. The spatial structure of urban system is the traditional field of human geography research. With the development of new urbanization and coordinated development of urban and rural areas, the research of urban system structure is developed from the morphological structure to social structure, cultural structure, flow structure, functional structure, and other fields [33]. Therefore, we need new perspectives and methods to support the study of the spatial structure of urban system.

The key to analyze the characteristics of user movement trajectories in specific areas is data acquisition. Traditional methods of group analysis in specific regions have three steps. First, define the group in the region; second, conduct a sample survey of the people in the region by using a questionnaire; third, analyze the statistical data. Traditional sampling survey could not get accurate facts to express data and always consumes large amount of time, manpower, and

material resources. Microblog data are the information released by microblog users in social networks. The content of microblog data has a strong real-time reliability and diversity, which has the advantage that traditional data do not have. In the case of scientific research, it is more accurate and persuasive to further analyze the user's information without revealing the user's personal information and privacy. Microblog data and traditional data have their own advantages. If we combine the two factors to analyze user behavior, the result of analysis will be more reasonable. The paper adopts this method when analyzing the correlation between microblog data and socioeconomic statistics.

The quality of data directly determines the accuracy of data analysis results. The data acquired by microblog should follow certain rules and should show its three characteristics of continuity, integrity, and validity. Therefore, when accessing data, we should pay attention to the time and number of interface calls, as well as the integrity and repeatability checking of data. The check-in information obtained in the study includes check-in city code, provincial code, check-in longitude and latitude coordinates, check-in date, and check-in time. Get microblog data and establish a database corresponding to the data field. After entering the warehouse, the missing data and the noise data are removed to ensure the completeness and uniformity of the data.

5.3. River Basin Perspective Study of Urban Spatial Interaction. In recent years, reunderstanding of the relationship between man and nature has become a hot topic in various fields. The city is an important habitat of human activities. The spatial and temporal structure changes of urban system imply the dynamic interaction between human and nature. Studying the spatial interaction of regional urban system and revealing the driving forces in the process of its change will help us to understand the temporal and spatial evolution of the urban system scientifically. The above research can provide the basis of space research for the sustainable development of human society and also provide a technical route reference for the spatial layout and optimization of the cities, which has a certain theoretical and practical significance.

As a complete physical and geographical unit, the River Basin crosses the administrative boundary and could reflect the regional human-land relationship more naturally and truly. The study of spatial interaction between cities by the river basin excludes the human subjective constraints, which more objectively reflect the spatial relationship characteristics of the regional urban system. Therefore, the river basin is an ideal area to explore the evolution rules of the relationship between mankind and land. The Huaihe River Basin, as the transition zone of China's urban system, has dual transitional nature and social elements [34]. Therefore, the research takes the Huaihe River Basin as a case area to study the spatial relationship of China's regional urban system. Its perspective is unique.

6. Conclusions

This paper obtains the user microblog information through the sina microblog open platform and studies the urban spatial pattern and urban interaction by means of statistical analysis and spatial analysis. This paper takes the Huaihe basin as the case area to verify it. The main research conclusion as follows:

(1) The data interface provided by microblog platform can study the urban spatial pattern. The user trajectory of microblog data can explore the spatial relationship of regional cities, and data acquisition and data quality evaluation can meet the research requirements.

(2) Based on microblog data, the spatial and temporal characteristics of urban system spatial pattern in Huaihe River Basin are analyzed from network connectivity and urban interaction. The study found that the urban spatial relation in the Huaihe River Basin has the following characteristics: the spatial difference of urban size distribution is obvious; urban layout presents a stratified aggregation phenomenon; and the high-grade cities lead the city's interaction.

As for the application of microblog data in urban research, the current mainly focus on information text, social relations, and other aspects. The research is mainly about event detection and hot spot exploration. The combination of big data thinking and data mining technology will have more research findings in the study of urban problems.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the Natural Science Foundation of China (Grant no. 41701187) and project funded by China Postdoctoral Science Foundation (Grant nos. 2018M640813 and 2018M633108).

References

[1] Chinese National Bureau of Statistics, 2017, in Chinese, http://www.stats.gov.cn.

[2] Y. Liu, L. Gong, and Q. Tong, "Quantifying the distance effect in spatial interactions," *Acta Scientiarum Naturalium Universitatis Pekinensis*, vol. 50, no. 3, pp. 526–534, 2014.

[3] Y. Chen, "Zipf's law, hierarchical structure, and shuffling-cards model for urban development," *Discrete Dynamics in Nature and Society*, vol. 2012, Article ID 480196, 21 pages, 2012.

[4] H. Zhang, L. Zhu, and S. Xu, "Modeling the city distribution system reliability with bayesian networks to identify influence factors," *Scientific Programming*, vol. 2016, Article ID 7109235, 2016.

[5] F. Zhen, B. Wang, and Y.-x. Chen, "China's city network characteristics based on social network space: an empirical analysis of sina microblog," *Acta Geographica Sinica*, vol. 67, no. 8, pp. 1031–1043, 2012.

[6] F. Giorgio and S. Gianluca, "Human-mobility networks, country income, and labor productivity," *Network Science*, vol. 3, no. 3, pp. 377–407, 2015.

[7] D. Dandan, Z. Chunshan, and Y. Changdong, "Spatial-temporal characteristics and factors influencing commuting activities of middle-class residents in Guangzhou city, China," *Chinese Geographical Science*, vol. 26, no. 3, pp. 410–428, 2016.

[8] W. Wu, Y.-h. Cao, S.-b. Liang, and W.-d. Cao, "The accessibility pattern of railway passenger transport network in China," *Geographical Research*, vol. 28, no. 5, pp. 1389–1400, 2009, in Chinese.

[9] J.-f. Xue, "Hierarchical structure and distribution pattern of Chinese urban system based on aviation network," *Acta Geographica Sinica*, vol. 27, no. 1, pp. 23–32, 2008, in Chinese.

[10] Z.-f. Jin, "On structural properties of transnational urban network based on multinational enterprises network in China: as the case of link with South Korea," *Acta Geographica Sinica*, vol. 29, no. 9, pp. 1670–1682, 2010, in Chinese.

[11] J. Yin, F. Zhen, and C.-h. Wang, "China's city network pattern: an empirical analysis based on financial enterprises layout," *Economic Geography*, vol. 31, no. 5, pp. 754–759, 2011, in Chinese.

[12] Z.–w. Sun, J.–l. He, and F.-j. Jun, "The accessibility and hierarchy of network cities in the global Internet," *Economic Geography (Chinese)*, vol. 30, no. 9, pp. 1449–1455, 2010.

[13] M. Stephens and A. Poorthuis, "Follow thy neighbor: connecting the social and the spatial networks on Twitter," *Computers Environment and Urban Systems*, vol. 53, no. 3, pp. 331–346, 2014.

[14] Y. Takhteyev, A. Gruzd, and B. Wellman, "Geography of twitter networks," *Social Networks*, vol. 34, no. 1, pp. 73–81, 2012.

[15] Z. Liu, Y. Jia, and X. Zhu, "Deployment strategy for car-sharing depots by clustering urban traffic big data based on affinity propagation," *Scientific Programming*, vol. 2018, Article ID 3907513, 9 pages, 2018.

[16] M. M. Rathore, A. Ahmad, A. Paul, and S. Rho, "Urban planning and building smart cities based on the Internet of Things using Big Data analytics," *Computer Networks*, vol. 101, pp. 63–80, 2016.

[17] D. Arribas-Bel, K. Kourtit, P. Nijkamp et al., "Cyber cities: social media as a tool for understanding cities," *Applied Spatial Analysis and Policy*, vol. 8, no. 3, pp. 231–247, 2015.

[18] J. Cinnamon and N. Schuurman, "Confronting the data-divide in a time of spatial turns and volunteered geographic information," *GeoJournal*, vol. 78, no. 4, pp. 657–674, 2012.

[19] A. Stefanidis, A. Crooks, and J. Radzikowski, "Harvesting ambient geospatial information from social media feeds," *GeoJournal*, vol. 78, no. 2, pp. 319–338, 2011.

[20] S. Elwood, "Geographic information science: emerging research on the societal implications of the geospatial web," *Progress in Human Geography*, vol. 34, no. 3, pp. 349–357, 2009.

[21] N. Ratnasari, E. D. Candra, D. H. Saputra, and A. P. Perdana, "Urban spatial pattern and interaction based on analysis of nighttime remote sensing data and geo-social media information," *IOP Conference Series: Earth and Environmental Science*, vol. 47, article 012038, 2016.

[22] A. Schulz, E. L. Mencía, and B. Schmidt, "A rapid-prototyping framework for extracting small-scale incident-related information in microblogs: application of multi-label classification on tweets," *Information Systems*, vol. 57, pp. 88–110, 2015.

[23] Z. Jiang, M. Evans, D. Oliver et al., "Identifying K Primary Corridors from urban bicycle GPS trajectories on a road network," *Information Systems*, vol. 57, pp. 142–159, 2016.

[24] L. Mitchell, M. R. Frank, K. D. Harris, P. S. Dodds, and C. M. Danforth, "The geography of happiness: connecting twitter sentiment and expression, demographics, and objective characteristics of place," *PLoS One*, vol. 8, no. 5, Article ID e64417, 2013.

[25] E. Steiger, R. Westerholt, B. Resch, and A. Zipf, "Twitter as an indicator for whereabouts of people? Correlating Twitter with UK census data," *Computers, Environment and Urban Systems*, vol. 54, pp. 255–265, 2015.

[26] Y. Shen and K. Karimi, "Urban function connectivity: characterisation of functional urban streets with social media check-in data," *Cities*, vol. 55, pp. 9–21, 2016.

[27] Z.-wei Sui, L. Wu, and Y. Liu, "Study on interactive network among Chinese cities based on the check-in dataset," *Geography and Geo-Information Science (Chinese)*, vol. 29, no. 6, pp. 1–5, 2013.

[28] G. Wessel, C. Ziemkiewicz, and E. Sauda, "Revaluating urban space through tweets: an analysis of Twitter-based mobile food vendors and online communication," *New Media and Society*, vol. 18, no. 8, pp. 1636–1656, 2016.

[29] M. Truelove, M. Vasardani, and S. Winter, "Towards credibility of micro-blogs: characterising witness accounts," *GeoJournal*, vol. 80, no. 3, pp. 1–21, 2015.

[30] J. Capdevila, M. Arias, and A. Arratia, "GeoSRS: a hybrid social recommender system for geolocated data," *Information Systems*, vol. 57, pp. 111–128, 2016.

[31] D. Kotzias, T. Lappas, and D. Gunopulos, "Home is where your friends are: utilizing the social graph to locate twitter users in a city," *Information Systems*, vol. 57, pp. 77–87, 2016.

[32] Chinese Weibo, 2018, in Chinese, https://weibo.com.

[33] Z. Wang, X. Ye, J. Lee, X. Chang, H. Liu, and Q. Li, "A spatial econometric modeling of online social interactions using microblogs," *Computers, Environment and Urban Systems*, vol. 70, pp. 53–58, 2018.

[34] Y. Fan, G. Yu, and Z. He, "Origin, spatial pattern, and evolution of urban system: testing a hypothesis of "urban tree"," *Habitat International*, vol. 59, pp. 60–70, 2017.