Information Extraction System for Transforming Unstructured Text Data in Fire Reports into Structured Forms: A Polish Case Study

*Marcin Michał Mironczuk** , National Information Processing Institute,
Warsaw, Poland

Received: 9 December 2017/**Accepted:** 15 July 2019

Abstract. In this paper, the author presents a novel information extraction system that analyses fire service reports. Although the reports contain valuable information concerning fire and rescue incidents, the narrative information in these reports has received little attention as a source of data. This is because of the challenges associated with processing these data and making sense of the contents through the use of machines. Therefore, a new issue has emerged: How can we bring to light valuable information from the narrative portions of reports that currently escape the attention of analysts? The idea of information extraction and the relevant system for analysing data that lies outside existing hierarchical coding schemes can be challenging for researchers and practitioners. Furthermore, comprehensive discussion and propositions of such systems in rescue service areas are insufficient. Therefore, the author comprehensively and systematically describes the ways in which information extraction systems transform unstructured text data from fire reports into structured forms. Each step of the process has been verified and evaluated on real cases, including data collected from the Polish Fire Service. The realisation of the system has illustrated that we must analyse not only text data from the reports but also consider the data acquisition process. Consequently, we can create suitable analytical requirements. Moreover, the quantitative analysis and experimental results verify that we can (1) obtain good results of the text segmentation (F-measure 95.5%) and classification processes (F-measure 90%) and (2) implement the information extraction process and perform useful analysis.

Keywords: Rescue report analysis, Text qualitative analysis, Text quantitative analysis, Information extraction system, Information system development, Information extraction, Firefighting report analysis

* Correspondence should be addressed to: Marcin Michał Mironczuk, E-mail: m.marcinmichal@gmail.com, marcin.mironczuk@opi.org.pl

1. Introduction

1.1. Background and Challenge Formulation

The National Fire Service (the Polish Fire Service) requires the creation of an electronic and paper documentation after each fire-brigade emergency (an intervention). A regulation [2] governs a form of documentation. The electronic version of the report is saved in an information system (database table, database, table, or data table) [10, 31]. The information system (IS) stores structured and unstructured information in the n -tuple form (an information structure or a data record in terms of a database system). The n -tuple is a sequence (an ordered list) of n elements, where n is a non-negative integer. We can consider the structured information as an n -tuple (*attribute, value(s)*) such as *((accidentType, forest fire), (numberOfVictims, 10), (temperature, 20))*. In this case, we do not consider the *values* as natural language text. The values *forest fire*, *10*, *20* are well defined and are derived from well-defined dictionaries such as the string set and real number set. Similarly, we can describe unstructured information in which *values* are described by natural language. For example, *(descriptionOfEvent, at the location of an accident involving two crashed cars: ford and bmw.)*. We can observe that the value *at the location of an accident involving two crashed cars: ford and bmw* is not derived come from well-defined dictionaries because it includes plain text.

Rescue brigades (firefighters) neutralise different hazards through various rescue activities. Information concerning such rescue service activities is stored on the database. Analysts in the Polish Fire Service use the electronic version of the reports to conduct various types of analyses. Unfortunately, they do not use unstructured information even though considerable interesting data can be extracted from the unstructured part of the reports. The analysts do not use the unstructured information because of the lack of appropriate tools and methodologies for analysing this information. Available traditional tools and methods process only the structured part of the reports. Therefore, a new challenge has emerged: How can we bring to light valuable information from the narrative portions of reports that escape the current attention of analysts?

1.2. Proposed Solution: General Overview

The main aim of this study is to present a solution that can transform unstructured data (the unstructured report or briefly referred to as the report) into structured information that can be processed to obtain valuable analysis and statistics. New information must satisfy certain business rules and the structure of that information must be an improvement of the old one. Figures 1 and 2 present an example of input data and output outcomes that describe the issues related to the transformation of unstructured data using an information extraction system.

Figure 1 presents *the hydrant* use case. As an input, the system receives the plain text that describes different aspects of a rescue action, and as an outcome, the system returns structured data in the form of a well-structured database record (d_1) that includes information concerning hydrants. Figure 2 presents *the*

(d_1 , After arriving at the accident site, I concluded that the balcony on the third floor was burning. The cabinets, wicker baskets, rugs, windows and façade were also on fire. In the offensive fire attack action two streams of water were directed on the fire: 1 from outside (of from the ground) on the balcony, 2 - from the staircase leading to the apartment. The doors were broken down. The smoke was released from the room, the site was submitted to the owner —. The car was tanked on the **Labiszynska** street, hydrant number **1673** - **efficient**.)

Unstructured input information,
i.e. a plain text (old information
structure)

Information extraction system

(d_1 , (**streetName**, Labiszynska),
(**id**, 1673), (**isEfficiency**, true))

Structured output information
(new information structure)

Figure 1. Information extraction from plain text—the hydrant use case. The system receives a report as an input and returns a structured record that describes different aspects of a hydrant, including the location (red colour), identifier (blue colour), and efficiency (orange colour), as an outcome (Color figure online).

car accident use case. As an input, the system receives another report, and as an outcome, the system returns a structured record (d_2) that describes a car accident.

1.3. Proposed Solution: Detailed Overview

In this article, the author proposes a novel information extraction system that analyses unstructured information and resolves the analytical drawback associated with the inability to analyse such information. The system enables the conversion of unstructured information into structured forms. Therefore, transformed information can be processed using analytical tools. Furthermore, the proposed platform includes components necessary for the realisation of information extraction (IE) from unstructured data sources (such as natural language text). These com-

Figure 2. Information extraction from plain text—the car accident use case. The system receives a report as an input and returns a structured record that describes different aspects of a car accident, including the types of vehicles that were damaged (red colour), types of rescue equipment used during the rescue operation (blue colour), and rescue activities involved during the rescue operation (orange colour), as an outcome. The acronym gcba 5/24 means the heavy firefighting car with a barrel (a water tank) and an autopump, where 5 means a water tank with a capacity of 5 m³ and 24 is a nominal capacity of the autopump 24 hl/min (2400 l/min) (Color figure online).

ponents are used in the identification, consequent or concurrent classification, and structuring of data into semantic classes, making the data more suitable for information processing tasks [29]. Figure 3 presents the main elements of the system.

The proposed solution in Fig. 3 is cascaded and includes qualitative and quantitative analysis. The author uses qualitative analysis to inspect the data acquisition and utilisation processes, including verification of the structure of information stored in the database system. This analysis helps in the development of suitable analytical requirements. A *Domain analysis* in Fig. 3 performs qualitative analysis procedures. The proposed quantitative analysis is based on five main steps, which include (i) text data pre-processing, (ii) taxonomy induction, (iii) IS

Figure 3. A schema of the system that performs information extraction from the rescue reports. The system consists of seven phases. Each phase delivers new information concerning a modelled domain and produces structured information that is used to perform new analysis.

schema construction, (iv) information extraction rule creation, (v) information extraction, and (vi) new analysis. Because of these steps, the data are presented in a structured form, which makes the realisation of useful analysis possible.

The *Text data pre-processing* phase includes two sub-processes, i.e. *text segmentation* and *semantic sentence recovery*. *Text segmentation* is responsible for splitting text into parts (segments) called sentences [3, 19, 35, 39]. Analysis of the reports can be a challenging task because the reports include many unusual abbreviations that may cause improper splitting of texts. The standard tools for text segmentation may provide improper results, and the author describes how this drawback can be resolved. *Semantic sentence recovery* is responsible for recognising the meaning of a sentence that is expressed as a label (a class name). Each sentence or groups of sentences from the report may describe different aspects of a rescue action, including weather conditions, types of equipment used, or rescue activities

involved. It is critical to set the appropriate label of sentences; owing to such categorisation, the search for relevant information can be enhanced and the analysis can be focused only on selected groups of segments. Furthermore, the author presents a comparison of the results of various text classification techniques [28, 37].

The *Taxonomy induction* step creates domain dictionaries [13, 17, 18, 30]. The dictionaries include significant terms from labelled sentences and contain information about the relations between such terms. For example, from the sentences that describe a car accident and the related rescue activities, crucial terms such as *police*, *emergency medical services*, and *the persons at the location of the accident*. In addition, dictionaries are used in the IS schema construction and information extraction. The processes involved in the creation of the dictionaries may be time consuming and require considerable human effort. However, this effort provides efficient results in terms of well-described domains (names of main domain concepts, names of attributes, relation names between attributes, basic set of attribute values) and reasonable information extraction outcomes. In addition, the article explains and presents the procedures involved in the construction of such dictionaries.

IS schema construction is the step that creates a physical database model [10]. This means that a database designer or a data analyst establishes, for example, for each table its names, names of attributes, types of attributes, and relations between the tables. Finally, a schema of the data record is also modelled. The author proposes using information from the *Taxonomy induction* step to support the modelling process. Moreover, the new schemas of information created are used to model the information received at the next information extraction step.

The next two steps, *information extraction rule creation* and *information extraction* are responsible for mapping information from sentences (unstructured information) into structures that contain structured information [29]. At the first stage, information extraction patterns are constructed. This means that an algorithm learns the ways of creating a pattern for recognising, for example, street names, street numbers, or hydrant types. At the second stage, the created patterns are used to extract the appropriate information, including specific street names, specific street numbers, or specific hydrant types. In the available literature, no related studies were found on domain specific terms extraction, which involves the extraction of terms from the firefighting area. For this reason, this article discusses the ways in which such processes may appear. In addition, the author presents the real information extraction process that has been implemented and utilised in the research performed.

The last step *new analysis* analyses the extracted and structured data. At this stage, we can create different statistics based on structured information. The author presents selected statistics from two use cases, a water point and a car accident. These statistics demonstrate the ability of the system to receive useful information that can be utilised to perform useful analysis.

1.4. Objectives and Novelty

The primary objectives of this article are outlined as follows: (1) This article presents a summary of the author's research on the application of an information extraction system in the field of rescue services. According to the author's knowledge, this topic has never been considered or widely explained. In addition, the article includes a unique source of descriptions of the problems encountered and the related solutions; (2) The author illustrates the novel use cases of the qualitative analysis methods (the failure mode effects analysis (FMEA) and the software failure tree analysis (SFTA) [16]) in the context of text data and information retrieval system analysis; (3) The paper explains the pre-processing step of text data, which includes sentence detection and sentence classification. Both steps are based on the author's solutions that improve both elements and outperform baseline approaches; (4) The author presents the process for taxonomy induction from text data. This solution is based on the proposed informal and formal analyses of text data; (5) The article describes the procedures involved in schema construction for the IS based on the created taxonomies; (6) The author discusses and explains the proposed methods for creating information extraction rules; (7) The article presents a sample analysis that can be created based on the information extracted.

Moreover, the two original use cases in this article are used to describe the proposed system. The first use case is related to the extraction of information concerning water points, which includes *the hydrant* use case. The second use case is related to the extraction of information about the car accidents, which includes *the car accident* use case.

1.5. Article Structure

The proposed study is practically well grounded, provides a deep understanding of the proposed solution, and is easy to understand. The paper is structured as follows. Section 2 presents the previous works. Sections 3–7 include detailed explanations of each stage of the information extraction system. Finally, Sect. 9 concludes the paper.

2. Previous Works

Conceptually, the proposed approach is closely related to frame theory [29]. However, the proposed method uses the information system (IS) concept rather than the frame idea. The author assumed that the IS concept is more appropriate and adequate in the presented case because it is frequently used for real implementation, and it is closely related to the database design. Furthermore, actual business processes in the Polish Fire Service use this concept. Owing to such reasons, the proposed solution is based on the IS concept.

The general overview of the system was presented in the article [27]. Nevertheless, the article does not explain the proposed solution in detail. In addition, a section related to semantic classification was introduced in the paper [25]. However, this article described only one element of the solution. Some analyses were intro-

duced in previous Polish articles and for this reason, they are not available to the broader research community. In addition, the previous papers explain only some parts of the proposed solution. In contrast to the above-mentioned articles, the present paper covers a comprehensive discussion of topics such as qualitative analysis of text data, text segmentation and classification, pattern creation, and analysis of final results. Moreover, the new analysis of *the car accidents* is introduced. The results of this analysis have never been published in any journal before. This article is a research summary of the analysis and utilisation of information from Polish Fire Service reports.

Nevertheless, it is worth mentioning three other works related to the author's proposition. The first work of Anderson and Ezekoye [8] proposes a methodology to supply estimates of the population protected by National Fire Incident Reporting System (NFIRS) [4]. The proposed methodology involves geocoding, i.e. a process that converting address data to some spatial coordinate system. The authors' proposition is very curious, and shows how we can utilise and process released public data to obtain a reasonable estimate of the protected population. However, the methodology based on the structured information, i.e. an address is given explicitly and have a well-defined schema. For this reason, the authors do not consider this information extraction approach directly to analyse the narrative information (a plain text) in contrast to author's proposition. The second solution is a RESC. Info Insight product [5, 6]. It is the attractive product that combines residential information, demographic information, and statistics from the fire department to insight which of factors plays an essential role in the likelihood of the fire at homes under consideration. This solution assumes that together with the abundance of data about the general population in residential areas, it should be possible to create insights in the locations where residential fires are more likely to happen. Unfortunately, there is a lack of any papers that describe how such models are created and how their work exactly. Unlike this product, the author describes and shows how we can utilise text data, and based on them create models and statistics explicitly. The last paper of A. Lareau and B. Long [20] presents the After-Action Review (AAR). AAR is a systematic analysis process and a comprehensive review of an operation that will identify strengths, weaknesses, and a path forward to an improved outcome on the next rescue event. This process is manual and has the form of a game where there are the players and team. For the AAR framework, we may try to adopt the collected unstructured and structured information for reasoning and considering past events and creating different calls/scenarios of emergencies.

3. Domain Analysis

3.1. General Overview

The rescue service domain consists of a varied range of institution activities and environments. We can analyse environmental properties such as regulations, acts, procedures, and databases. In addition, we can analyse the existing relations between these properties. We perform domain analysis to obtain requirements.

These requirements explain what must be done to improve the analysed domain. The analysed area might become more robust to errors and changes owing to the implementation of requirements. Generally, in the first step of the analysis, we can detect several environmental defects by analysing the environment of the domain. In the second step, solutions in the form of requirements can be proposed to resolve the defects detected. The new requirements are usually associated with changes in procedures, law, and data tables.

A contribution of this article includes the illustration of qualitative techniques such as FMEA and SFTA, which can be used to analyse a current database and create new requirements for the new database [16]. FMEA is a systematic process that predicts the causes of errors in a machine or a system and estimates their significance. SFTA is executed to predict system errors in advance rather than to prove that it works correctly after the creation of or during test procedures. These methods can determine (1) the failure of the current database in some cases (e.g. why cannot we obtain the appropriate information through information retrieval?); and (2) where problems concerning the implementation of new information processes will occur (e.g. why cannot a system realise new requests for some information or new business rules/ use cases?). Furthermore, owing to these methods, we can determine whether something has been omitted or has not been considered in the design process. This type of analysis can confirm or falsify the possibility of using the current database to implement new information processes or business rules. In addition, in the case of falsification, the analysis provides a new schema of information (a new data table) that must overcome and eliminate the disadvantages of the prior information structure. The practical preliminaries of this topic are presented in “Appendix A”.

3.2. Running Example

The National Fire Service electronic and paper documentation is created for every fire brigade emergency. The electronic version is stored in the database table. Commanders of a rescue operation use natural language to describe several aspects of a rescue action in the *Descriptive data for information of an event* field of paper documentation. *The descriptive data for information of an event* field is divided into six subsections, namely (1) *description of emergency actions (hazards and difficulties or worn-out and damaged equipment)*, (2) *description of units that arrive at the accident site*, (3) *description of what was destroyed or burned*, (4) *weather conditions*, (5) *conclusions and comments arising from the rescue operations* and (6) *other comments about data on the event that has been filled in the form*. After the paper documentation is complete, its electronic version is saved as a data record in the table.

3.2.1. Environment Analysis: Text Data Analysis Figures 1 and 2 present real reports (unstructured input information), i.e. real descriptions of the rescue actions from the database table. As we can observe, during the process of mapping information from the paper to an electronic document, semantic information is lost, which implies that the meaning of the sentences is lost. The current

table does not include such division in the subsections mentioned above. When a single report is written, semantic information is lost, which limits the gathering of information from the appropriate subsections or searching for particular paragraphs.

Based on 4000 evaluated reports, the author established and confirmed that the contents of the reports can be classified into five categories (semantic classes). The heuristic rules are used to manually assign sentences (the segments) of a report to these categories. An example heuristic rule works as follows *if the sentence contains the words associated with damage, then classify this sentence as a damage class*. Consequently, a set of reference sentences (SORS) has been obtained. The received names of classes are identical to the paper documentation classes. These classes may include segments that construct an event description. The author established the following class names *operation*, *equipment*, *damage*, *meteorological condition*, and *description*. Table 1 presents the semi-structured report i.e. the report is divided into the classes mentioned above.

Table 1 presents the results of the division of the report from Fig. 1 in five sentences in which each sentence assigns one semantic label. This procedure helps recover the meaning of the sentences, and it is important for further processing of the sentences (the classification and extraction processes).

3.2.2. Environment Analysis: Information System Analysis After the analysis of the reports, the FMEA of the unstructured and semi-structured reports (Table 1) was performed. The semi-structured report includes semantic sections (the report divided into the semantics sections). The unstructured report does not include such information. The author created four scenarios for information retrieval from the unstructured and semi-structured reports. The information was retrieved by using a query *q* (a full-text search). In some cases of the search, it was assumed that not only text fields would be available in the search system. It was assumed that some meta-data such as the time of a rescue action or its location would also be avail-

Table 1
Example of a Semi-Structured Report [25]

Sentence	Class name (Semantic label)
After arriving at the accident site, it was concluded that the balcony on the third floor was burning. The cabinets, wicker baskets, rugs, windows, and façade were also on fire.	Description
As an offensive fire attack action, two streams of water were directed to the fire: 1, on the floor of the balcony; 2, on the staircase leading to the apartment.	Operation
The doors were broken down.	Operation
The smoke was cleared from the room, and the site was handed over to the owner —.	Operation
The car was tanked on Labiszynska street, hydrant number 1673—efficient.	Equipment

The report is divided into five sentences, and each sentence is assigned a semantic class label

able. Following this, the query q was sent to the search index that was constructed based on unstructured and semi-structured reports (the unstructured search index and the semi-structured search index). The query excludes or includes some information. In other words, q excludes the semantic section and excludes the meta-data; q excludes the semantic section and includes the meta-data; q includes the semantic section and excludes the meta-data; q includes the semantic section and includes the meta-data. The query results were noted for each instance. Alternatively, the effects of such searches were predicted. The FMEA performed on the two structures of the reports explained how these systems fail in the case of hypothetical scenarios. For example, it was determined that when the query q - *hydrants in area X* is sent to the unstructured search index, the results produced might not be relevant. In this case, the results will contain useless information, i.e. the descriptions of all actions in the area of X . This is not the same as in the case where we send a similar query to the semi-structured search index. The results of this analysis were finally used to design an improved solution, which comprises a new structure of information (a new schema of a database table) that describes structured information (structure output information in Figs. 1 and 2).

Following the FMEA analysis, the design process of the new database table was initiated. This process begins with an SFTA analysis. Figure 4 presents the results of this analysis.

Figure 4 presents the SFTA analysis that provides a list of critical paths. The unstructured search index exhibits several flaws that demonstrate that such an index cannot be used to obtain appropriate and precise information. The index compiled from the unstructured reports returns too much or too little information, and consequently, rescue commanders and experts cannot find useful information and take the right decisions. The primary cause of this flaw is the fact that information is stored in a textual form and the retrieval system returns imprecise and ambiguous results. The semi-structured search index partially addresses the problems associated with an unstructured search index. Consequently, queries to the appropriate section of rapports that contain the necessary information can be limited. Unfortunately, the results are presented in plain text form and analysis of the results to extract precise information may still be time consuming. For this reason, a new and improved structure of information is required.

3.2.3. Requirement Formulation The new structure of information must deactivate the critical paths illustrated in Fig. 4. Consequently, the new solution does not disappoint users when they search for information. Based on the analysis performed, the semi-structured form of the reports and the target version of the information structure were established as new solutions to the problem of information retrieval in the current unstructured index. The semi-structured search index was proposed as an intermediate solution that stores semi-structured reports (see Table 1). In addition, the index includes data that are required by the new structured index. The structured index stores the final information, which includes new structured information about *water points* and *car accidents*.

Figure 4. A graphical representation of the SFTA of the database table [27]. The figure presents the primary paths and reasons that cause the primary effect, i.e. the required information is not obtained.

3.3. Summary

In summary, the most important findings in this step are as follows: (1) The current unstructured index fails when we use the information retrieval approach for obtaining precise information (the requirement). The system fails because this requirement has been omitted in the design process and the data structure (text data) is not appropriate for fulfilling this requirement; (2) The proposed analysis based on FMEA and SFTA falsify the possibility of using the unstructured data to realise new business processes, i.e. precise information retrieval (the unstructured index contains the required information but it can be difficult to access); (3) The analysis performed provides the requirements and a non-formal solution, i.e. the intermediate and final new schemas of information may resolve the informa-

tion retrieval problem. The new structure of information must overcome and eliminate disadvantages of the former structures.

4. Text Data Pre-processing

4.1. General Overview

The text data pre-processing process receives plain text as an input and returns the semi-structured report as an outcome. The process includes two main components: sentence detection and text classification.

The sentences/ segment boundary detection problem [3, 19, 35, 39] is a fundamental problem in natural language processing (NLP), which is related to determining the location where a sentence begins and ends. The problem is non-trivial because while some written languages have specific word boundary markers, in other languages the same punctuation marks are often ambiguous. For example, a segmentation tool uses input received as plain text (see Fig. 1) and returns a list of sentences (see Table 1, column *Sentence*). Further, text classification or categorisation is a problem of learning a classification model from training documents labelled by pre-defined classes. Such a model is used to classify new documents [7, 9, 22, 28]. For example, a classification model receives a sentence (see Table 1, column *Sentence*) as an input and returns a label/ class name (see Table 1, column *Class name*) as an outcome.

Contributions of this part of the article include a presentation of the constructed solution for sentence detection, and this solution outperforms alternative methods. In addition, this demonstrates that we can achieve satisfactory classification results using the sentences.

4.2. Running Example

4.2.1. Sentence Detection In the presented case, reports are characterised by numerous anomalies in spelling, punctuation, vocabulary, and a large number of abbreviations. Despite these obstacles, an appropriate solution to text segmentation was developed [26]. This tool uses two knowledge databases. The first database describes several types of common abbreviations. The second database includes information on patterns (rules) that indicate the start and end locations of a sentence. Owing to these databases, the segmentation tool produces better results than approaches such as the segmentation rules exchange (SRX) or solutions obtained from open source projects related to NLP [1, 23]. Figure 5 presents a comparison of results obtained.

Figure 5 presents a comparison of histograms of the segmentation tool results. Each chart in Fig. 5 compares the reference data set that includes sentences with the results obtained from each segmentation tool. We can observe that the proposed solution and SRX provide the same set of reports that include a set of same-length sentences. The NLP tool produces excessive sentences. Furthermore, the proposed solution better fits the reference data set than other alternative solutions.

Figure 5. Comparative histograms of the results obtained using the segmentation tools [26]. A collection of reference segments is compared with three other sentence datasets obtained from solutions, namely SRX, openNLP, and the proposed segmentation tool.

4.2.2. Sentence Classification

4.2.2.1. Dataset preparation Following the segmentation of reports and the manual labelling of sentences, SORS was obtained. Manual labelling is a step required for supervised learning [37]. The SORS contains nearly 1200 sentences classified into five different classes. Based on the SORS, we can obtain the following statistics for the labelled segments in each class: (1) the operation –37%, (2) the description –31.9%, (3) the equipment –16.9%, (4) the damage –7.4%, and (5) the meteorological condition –6.8%. We can conclude that commanders frequently describe different rescue activities, i.e. they describe resolved rescue problems in detail. They rarely describe the damage and the meteorological conditions at the location of the accident.

4.2.2.2. Classification process Following the segmentation process and the sentence labelling phase, a tool for recognising the semantics of sentences was implemented. This tool uses a supervised classification technique to learn a model that classifies sentences into one previously described semantic classes (a multi-class

Figure 6. The best precision and recall obtained from the classification process (the binary weight of features) [25]. The X-axis indicates precision and the Y-axis indicates recall. The left diagram presents results in the case involving the lemmatisation process. The centre diagram presents results that used the n -grams representation of text for classification. Classification results from all feature spaces are presented in the right diagram.

classification problems¹) [25, 28, 37, 38]. In the previous research, the author conducted experiments in this field. Different machine learning methods such as k -nearest neighbours (k -nn), naive Bayes (NB), Rocchio and its author's modification were tested [7, 9, 22, 25, 36]². Figure 6 presents the example results of the classification performed.

Figure 6 illustrates the best coefficients, i.e. precision and recall³ obtained from the classification process using classifiers such as k -nn, NB, Rocchio and its modifica-

¹ A multi-class classification problem models a classification problem with more than one class where each sample (sentence) is assigned to one and only one label.

² A k -nn classifier assigns a class label of the k closest training examples to a new example; a NB classifier sets a class label to a new example based on applying Bayes' theorem with strong (naive) independence assumptions between the features/ terms; a Rocchio classifier creates for each class a centroid (the vector average or centre of mass of its members) and classifies a new example to the nearest centroid. The author's modification of the Rocchio classifier takes into account meta-information, such as a sentence position in text and sentence length.

³ Precision is the proportion of sentences predicted positive that are actually positive, whereas recall is the proportion of positive sentences that have been correctly predicted as positive [28]. The F-measure is a harmonic mean of precision and recall, i.e. $F - measure = (2 \cdot precision \cdot recall) / (precision + recall)$.

tion. A diagram in the middle of Fig. 6 demonstrates that when we use the n -grams, i.e. a sequence of n adjacent words from the sentence, such as *car accident* or *burned cabin* [11], the classification results are worse when compared to the other two cases mentioned below. The lemmatisation process (the left diagram in Fig. 6) affects the NB classifier in relation to the classification of all feature sets/ all single words, i.e. one-grams (the right diagram in Fig. 6). In this case, the F-measure increases by 2%. Furthermore, the best classification results are obtained by the k -nn classifier.

4.3. Summary

In summary, the most significant findings in this step include the following: (1) The proposed sentence detection solution outperforms the baseline approaches. A better fit on the reference set was achieved; (2) The domain texts required different segmentation rules than typical text; (3) Machine learning algorithms were used to recover semantics sections from reports; (4) The proposed modification of the Rocchio classifier outperforms the classification results, but the best results are achieved when the k -nn classifier was used; (5) The lemmatisation process significantly reduces the feature number so that a classifier outperforms others with similar processing requirements; (6) The semi-structured index in which semi-structured data is saved satisfies part of the improved data representation requirements, i.e. reconstructed / labelled semantic sections are obtained.

5. Taxonomy Induction

5.1. General Overview

Taxonomy induction is the step in which semi-structured reports (sentences that belong to a semantic class) are analysed. In addition, taxonomies are created in this phase. The term *taxonomy* refers to the classification of items or concepts, including schemes that describe such classification. In the study context, taxonomy relates to the orderly classification of terms and phrases according to their presumed natural relationships and groups.

A contribution of this part of the article is the proposition of a two-step technique for taxonomy building from the reports. First, an informal approach is used. This approach utilises mind maps to determine the primary attributes of a new information structure. In the second step, the formal attitude based on formal concept analysis (FCA) is utilised [34, 40] (“Appendix B” contains a general overview of FCA). However, the author did not find any related studies on the use of such techniques for the analysis of rescue reports. For this reason, the author proposes an analysis of the reports.

5.2. Running Example

5.2.1. Informal Analysis The required sentences are first obtained from the semi-structured reports. The sentences labelled as *equipment* that contain a term *hydrant* for the *hydrant* use case are obtained. All sentences of the reports that contain information concerning the *car accident* for the *car accident* use case are

Figure 7. (a) Shows an example of a mind map of the hydrant concept [27]. (b) Presents an example of a mind map of the car accident concept. Both mind maps contain the major terms/ attributes that were extracted from the sentences.

collected. The manual analysis is then performed. During the analysis, each sentence is read, and potential attributes are manually extracted. In addition, the derived attributes from the reports can be used for consultations with a domain expert. Following this, a hierarchy of attributes in the form of a mind map is designed. Figure 7a presents the mind map for the hydrant concept and Figure 7b presents a particular part of the mind map for the car accident concept (the entire taxonomy is available in the data repository).⁴

The created taxonomies, which are presented in Fig. 7a, b, enable the (1) recognition of the vocabulary available in the explored field, (2) extraction of pre-attributes, and (3) creation and visualisation of the fundamental relationship between attributes. Furthermore, the taxonomies comprise the first dictionary of attributes that are used in the formal analysis step.

⁴ <http://doi.org/10.5281/zenodo.885436>

Table 2
Example of the Simple Data for Building the hydrant Formal Context

The objects	The attributes		
	c_1 = location	c_2 = inefficient hydrant	c_3 = causes of inefficient
o_1 = <i>labszynska street</i>	1		
o_2 = <i>do not work</i>		1	
o_3 = <i>broken off bolt</i>		1	1

Rows contain extracted phrases from sentences and columns contain label names. When a label fits in a phrase, a cell is then set as 1; otherwise, a cell is empty

5.2.2. Formal Analysis When the first version of the taxonomies is established, more formal and precise versions of the taxonomies can then be created. The author proposes a semi-supervised method for creating such taxonomies. Phrases from the appropriate sentences are extracted for this purpose and are then manually labelled with the appropriate attributes. In addition, attributes (phrases) from previously created taxonomies are used. The phrases of a broader context are used as the names of attributes. Table 2 presents a simple example result of such an operation.

In Table 2, the rows contain phrases extracted (objects in the term of FCA) from the sentences, and the columns include attribute names (attributes in the term of FCA). If a given attribute's name covers a phrase semantically, i.e. there is a semantic relation between an object and the attributes (for example, *labszynska street is a location*), then we set 1 in the appropriate cell; otherwise, an empty value is set. Following this, the FCA algorithm uses data from Table 2 to generate a lattice. Figure 8a, b present the created lattices for *the hydrant* and *the car accident* use case, respectively.

Figure 8a presents a lattice for *the hydrant* use case. The generated lattice contains concepts (circle symbols in Figure 8). Concepts aggregate attributes and their values (objects in the FCA convention). For example, a concept labelled as *hydrant* aggregates an attribute of that name and its values, i.e. *overwhelmed hydrant*, *underground hydrant*, *frozen hydrant*. This concept is separated into several sub-concepts, i.e. *hydrant type* and *causes of an inefficient hydrant*. The concept *hydrant type* includes values such as *above-ground* and *underground*. The concept *causes of an inefficient hydrant* includes values such as *overwhelmed hydrant*, *frozen hydrant*.

Figure 8b presents a lattice for *the car accident* use case. In this case, the author modelled only some part of *the car accident* domain. In this lattice, for example, a concept labelled as an *internal resource* aggregates the attribute of that name and its values, e.g. *brok*, *amelin*, *no. 6 warsaw*. This concept is divided into several sub-concepts, i.e. *vfb* (*the voluntary fire bridge*) and *rafu* (*the rescue and firefighting units*). The concept *vfb* includes values, such as *brok* and *amelin*. The concept *rafu* includes values, such as *no. 6 warsaw*, *lipno*. The *brok* and *amelin* are the proper nouns of *the voluntary fire bridge*, and *no. 6 warsaw* and *lipno* are the proper nouns of *the rescue and firefighting units* respectively.

Figure 8. (a) Shows a lattice of the hydrant concept [27]. (b) Presents a lattice of the car accident concept. Both lattice orders and the major terms/ attributes were extracted from the sentences.

5.3. Summary

In summary, the most important findings in this step are as follows: (1) The proposed informal analysis allows the recognition of the domain vocabulary and the relations between the main attributes; (2) FCA can be used to create more precise and complex taxonomies that cover not only the relations between attributes but also include example values of the attribute (the hierarchy of the dictionaries); (3)

The created lattices visualise the main relations among the main domain attributes in a simple and clear way.

6. Schema Construction

6.1. General Overview

Schema construction is a stage in which the taxonomies are analysed and a physical data model (a new structure of information) is created. The physical data model demonstrates the manner in which the model is built in the database. The model illustrates the following: table structures, including column/ field names, column/ field data types. This is a technical step in the proposed analysis, but it is very important because it delivers the final schema of data utilised in the next information extraction step. “Appendix C” describes, in detail, the data structures created for *the water points* and *the car accidents* use cases.

6.2. Summary

In summary, the most important findings of this step are as follows: (1) We can create the physical data model of new information from the constructed taxonomies; (2) The designed model fulfils data representation requirements that allow for more precise information search; (3) The designed model covers all required attributes for which values might be extracted from the available text data.

7. Creation of Information Extraction Rules and Information Extraction

7.1. General Overview

Creation of information extraction rules and information extraction are two mutually dependent stages. First, rules (information extraction patterns) are developed on the basis of text data. Next, the created rules are used to extract new information from text. “Appendix D” presents practical preliminaries related to information extraction.

In the considered case, we focus on rules for recognising the names of entities. This process is referred to as name entity recognition (NER) [29]. Here, the NER recognises and classifies named expressions in text, for example, people’s, cars’, locations’ and equipment names. For example, let us consider the following unstructured n-tuple (d_1 , *Jan Smith drive a GCBA 5/24.*). The n-tuple includes a document id as a key and text as a value. The NER algorithm receives the text (*Jan Smith drive a GCBA 5/24.*) as an input and returns two different n-tuples such as (*person*, *jan smith*) and (*fire-truck*, *gcba 5/24*) as an output. Following this, the keys and values are mapped into the appropriate fields of a new information structure.

This part of the article describes and explains how we can realise such information extraction processes from rescue reports. In addition, this section presents the author’s solution to the task of name entity recognition.

7.2. Running Example

7.3. Use Case: Water Points

In this use case, the author manually created a set of extraction rules (see a description of a manual pattern induction in “Appendix D”). The orderliness of these rules is achieved in an unsupervised way using the FCA. The set of rules is the order of the set of pairs (A, B) of the formal context, where A is a set of sentences and B is a set of pairs (a, p) that cover the sentences from A . The pair (a, p) implies that a given sentence from A is covered by the extraction pattern p , which relates to the appropriate attribute a of a new information structure. Using this pattern, we can extract values of a given attribute from the sentence and map them into a new information structure.

The author analysed 1,523 sentences and established 19 information extraction patterns, for example, *phrase hydrant pattern street name street number*, *pattern street name street number*, *phrase number pattern id*, *pattern hydrant type* [24]. Each sentence is linked to relevant pattern(s) and the lattice is created. Figure 9 presents selected information extraction patterns organised in the lattice form.

Figure 9 presents a part of the created lattice. This lattice indicates the relations between concepts with aggregate information on sentences and the related extraction patterns. Each concept of the lattice includes *(an attribute name of a new information structure and the pattern)* pairs. The attribute values of the new information structure can be extracted based on the extraction patterns in the sentence.

The following example illustrates the operation of NER based on the above rules. Let us analyse the following sentence $d_p = \text{hydrant hoza 30 number 140 deep, overwhelmed}$. All patterns from concept c_{11} (Fig. 9) have been paired with the

Figure 9. Part of a lattice that describes the relations among information extraction patterns. Each concept of the lattice contains the pattern combinations that cover sentences.

parts of the sentence. The attribute *phrase hydrant pattern street name street number* covers *hydrant hoza 30* part of the above sentence and the *pattern street name street number* extracts a *hoza 30*. (*hoza* - is the street name, *30* - is the street number). The next attribute *phrase number pattern id* covers number *140* part of the above sentence and the *pattern id* extracts *140*. The attribute *pattern hydrant type* covers *deep* part of the sentence and the pattern *hydrant type* immediately extracts *deep* value (*deep* - is a synonym of the underground). The attribute *pattern hydrant description* covers *overwhelmed* part of the sentence and *pattern hydrant description* immediately extracts *overwhelmed* value. If the above sentence does not contain, for example, *overwhelmed*, then the extracted rule from concept c_8 (Fig. 9) is used. Because of these operations, we receive the following n-tuple $(1, (id, 140), (isLabeled,), (types, \{deep\}), (isWorking,), (defectivenessReasons,), (description, overwhelmed), (lat, 52.22), (lon, 20.939), (streetname, hoza 30), (objectName,), (locationPhrases,))$, where the values of the *lat* and *lon* attributes are obtained through a geotagging service.

7.4. Use Case: Car Accidents

The following use case is less complex. In this case, the author used the manually created dictionaries of attribute values to extract information (see the description in “Appendix D”). Dictionaries are first defined, for example, $V_{vehicles} = \{Peugeot\ 205, Ford\ Mondeo\}$, $V_{equipment} = \{gcba\ 5/24\}$, and $V_{operations} = \{cut\ the\ electricity, cleared\ the\ accident\ place\}$. Second, each value from the dictionaries is matched to the sentences. For example, if we match the values from the dictionaries to the plain text in Fig. 2, we obtain the following n-tuple $(1, (id, 1), (vehicles, \{Peugeot\ 205\ Ford\ Mondeo\}), (equipment, \{gcba\ 5/24\}), (operations, \{cut\ the\ electricity, cleared\ the\ accident\ place\}))$

7.5. Summary

In summary, the most significant findings of this step include the following: (1) We may utilise the FCA to analyse information extraction patterns and use a created lattice to implement an information extraction solution that is based on hierarchically organised extraction rules. It is a novel approach to analysing fire reports; (2) The hierarchy of extraction rules covers all possible combinations of patterns that occur in the analysed sentences. This implies that all available information from the sentences can be extracted; (3) The dictionary approach to information extraction is a simple solution and a useful approach. However, the dictionary approach requires human effort and in complex cases, may be inefficient and impractical; (4) The new information structure fulfils the previously created requirements and enables a more precise information search.

Figure 10. Classification of hydrants based on their location [27]. Each node of the tree includes information on the number and percentage of cases that belong to a given category.

8. Analysis

8.1. General Overview

An analysis is a step in which statistics in the form of tables or charts are created based on structured information. These statistics help acquire information and gain knowledge about an analysed domain. This part of the article presents the useful analysis developed.

8.2. Running Example

8.3. Use Case: Water Points

In this subsection, the author presents a simple statistic of hydrant locations (Fig. 10). Other statistics of water points are presented in “Appendix E.1”.

Figure 10 presents a division of hydrants based on their location. It can be concluded that about 69% of sentences include location information. In addition, this collection can be divided into sentences that describe hydrants located near the streets (77.84%) and at the crossroads (22.16%).

8.4. Use Case: Car Accidents

For the *car accidents* use case, the author presented statistics on motor vehicles involved in car accidents (Fig. 11). Additional analysis is provided in “Appendix E.2”.

Figure 11 presents a frequency diagram of motor vehicles involved in car accidents. The author found nine different, most frequently occurring motor vehicles in 205 selected reports. There are 205 records and nine unique names of motor vehicles (nine unique values of *vehicles* attribute, see Fig. 12b). It can be concluded, based on the analysis of Fig. 11, that the following motor vehicles are most frequently mentioned (*frequency*>2%) in the reports: trucks (7.8%), volkswagen golf (6.34%), volkswagen (4.88%), BMW (2.93%), opel (2.93%), opel astra (2.93%), polonez (2.93%), opel corsa (2.44%), and passat (2.44%). More-

Figure 11. Motor vehicles involved in car accidents (the motor vehicles frequency > 4%). Figure presents a number of reports and a percentage of chosen motor vehicles involved in car accidents.

over, we can obtain information on frequency patterns, i.e. information concerning what motor vehicles were involved in car accidents. The author extracted the most frequently occurring combination (*support* $\approx 1\%$, *confidence* $\geq 70\%$) patterns of motor vehicles, e.g. (alfa romeo, opel astra), (trucks, volvo), and (passat, volkswagen).

8.5. Summary

In summary, the most significant findings in this step include the following: (1) The new information schema created helps control the amount of information and the convenient and accessible presentation of such information, which fulfils the established requirements; (2) Analysis based on structural information may enable decision makers to prepare better rescue operation plans and manage emergency equipment.

9. Conclusions

This article described a framework for systematic information extraction from fire reports. Moreover, the author demonstrated real problems that occur during framework implementation and proposed the relevant solutions. Furthermore, the implementation performed has been supported by two real use cases and data obtained from the Polish Fire Service. Such a topic has not been previously considered and has not been widely and comprehensively described in the rescue literature. In addition, analysts from the fire service have had no opportunity of using text data for analysis. Currently, two drawbacks have been addressed in this study and the proposed information extraction system. Owing to the described proposition, it is possible to reuse text data and for fire service analysts to conduct new analysis. Moreover, the analysis performed helps analysts understand the following issues, namely: (1) the process of data acquisition, its drawbacks, and how we can eliminate the disadvantages and obstacles of this process; (2) the sentence detection and classification tasks, and how they allow us to acquire the relevant sentences that describe different aspects of rescue actions; (3) the vocabulary included in the reports thanks to the proposed solution of taxonomy induction; and (4) the process of information extraction, and how we can acquire structural information and create a more precise query for searching relevant information/ facts that may enable decision makers to prepare better rescue operation plans and manage emergency equipment.

Also, the article demonstrates the novel use of qualitative and quantitative analysis techniques. First, the author has shown that the failure mode effect analysis can be used to determine drawbacks in the reports and the current search system, and the system failure tree analysis can provide critical paths of such systems. Both analyses have made contributions to the development of improved solutions. Second, the article described three novel uses of formal concept analysis. The first use case demonstrates the ways of creating tools for sentence detection. The second use case illustrates the ways of inducing taxonomies from rescue reports. The last use case describes the design problem of a solution concerning information extraction. Furthermore, the article has demonstrated that sentence classification can be performed satisfactorily, resulting in helpful analysis. Also, it is worth mentioning that the proposed system and methods are language independent. The methods use only sequences of tokens and do not use any special language rules or grammar and syntax layers to resolve the enumerated problems. In conclusion, the proposed process is complete, not trivial, and requires considerable effort. The

author has demonstrated that the presented approach adequately resolves the formulated problem.

Acknowledgements

The authors would like to thank the editor and the reviewers for the time, effort, and valuable comments that they have contributed to this study.

Open Access

This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

Appendix A: Domain Analysis: Practical Preliminaries

Database engineers or architects usually create a database schema. Most often, this schema is developed on the basis of an external world analysis, i.e. the environment of a modelled domain. Data tables are designed, and the designed tables are filled with data. Because of this, it is possible to perform established information processes so that business rules can be realised. Usually, after specific time information requirements of the system’s end-users are changed, there exists additional requirements for new types of information or analysis, i.e. new business rules are created. Optionally, new unplanned use cases of data tables are created. It can sometimes be difficult to access some particular information from the table.

Figure 12. (a) Presents a data table schema of the hydrant. (b) Presents a data table schema of the the car accident. Each schema contains the name and the set of attributes with types.

For example, the designer of the table established a textual representation of attribute values.

In such a case, the information required is available in a table but can be hard to access. Obtaining this information often requires additional transformations such as data merging and data exploration. These processes are generally time consuming. For such reasons, refactoring the data schema is required, and a new data schema must be created. The new information structure can be obtained through induction - the old information structure can be redesigned and restructured into the new information structure. For this purpose, we can use the refactoring process, and we can utilise an information extraction system that enables this transformation.

However, some factors complicate the refactoring process. These factors include the lack of knowledge concerning older design processes of the table (assumptions, specifications) and the lack of knowledge concerning the exact structures and contents of the table. There are usually only specific pieces of data that contain the attributes and values in the data table. There is no access to all attributes and values that are incomplete or inaccurate. Despite these limitations, we can determine the drawbacks of the table and propose better solutions based on qualitative performance analysis, including FMEA or SFTA related to the design for trustworthy software (DFTS) method [16].

Appendix B: Taxonomy Induction: General Overview of FCA

Rudolf Wille introduced FCA in 1984. FCA is based on a partial order theory invented by Birkhoff in the 1930's [15, 32–34, 40]. FCA serves, among others, to build a mathematical notion of a *concept* and provides a formal tool for data analysis and data representation. Researchers use a *concept lattice* to visualise the relations between the discovered concepts. A Hasse diagram is another name of the concept lattice. This diagram consists of nodes and edges. Each node represents a concept, and each edge represents a generalisation/ specialisation relation. FCA is one of the methods used in knowledge engineering. Researchers use FCA to discover and build ontologies (for example, from text data) that are characteristic to particular domains [12, 14, 21, 32].

Appendix C: Schema Construction: Running Example

Schema construction is a process where table and data record schemas are created, i.e. names of tables and attributes are established (physical data model). Figure 12a, b present the designed schemas for the *hydrant* and the *car accident* use cases, respectively.

Figure 12a, b present the final schemas (the interfaces) of the manually created data tables. We use these interfaces to create the appropriate instances of extracted information (a data record). We can observe that most of the attributes of these interfaces are the same as the attributes of the created lattices from Fig. 8a,

b. The lattices mentioned above help us choose the attribute and class names and establish relations between these elements for the modelled domain.

The *Hydrant* class (Fig. 12a) is the table schema that contains the attributes such as *id* (a hydrant identifier), *isLabeled* (a boolean flag indicates whether hydrant had a nameplate), *types* (a list of phrases, such as *underground*, *ground*, indicating the type of the hydrant), *isWorking* (a boolean flag indicating whether a hydrant was efficient), *defectivenessReasons* (a list of phrases explaining the defectiveness reasons), *description* (a text describing the environment (place) where the hydrant was located, such as *cars blocked the hydrant*, *the frozen snow blocked the hydrant*), *lat* (latitude of the hydrant), *lon* (longitude of the hydrant), *streetName* (a street name where the hydrant is located), *objectName* (a text that provides the location context related to physical objects, like *bus station*, *hospital*, etc.) and *locationPhrases* (a text that provides the location context related to abstract objects, like *cross roads*, *traffic circle*).

The *CarAccident* class (Fig. 12b) is a table schema that contains the attributes such as *id* (identifier of the car accident), *vehicles* (phrases describing marks of the vehicles, e.g. *bmw*, *opel*), *rescueUnits* (phrases describing types and names of rescue units, e.g. *brok*, *amelin*), *equipment* (the type of equipment that was used in the accident site, e.g. *gba*, *sop*, *sawdust*), and *operations* (phrases explaining the type of operations performed at the accident site, e.g. *evacuation of people*, *cut off electricity*).

Appendix D: Information Extraction: Practical Preliminaries

Figure 13 presents the induction techniques used to extract patterns of each attribute value for the information extraction task.

According to Fig. 13 and based on the input text, each attribute's value can be induced by (1) manual review of each value of the selected attributes of the table and transfer the relevant parts to the relevant attribute(s) of the new data table (manual matching, manual data extraction, and data mapping); (2) manual creation of dictionary (attributes are created, and closed sets of values are assigned to them), the values from the dictionary are matched with values from the table. The matched values are then transferred to the corresponding attribute(s) of the new table as new values (an automatic matching and data extraction and manual or automatic data mapping); and (3) manual creation of a dictionary of patterns (the attributes have assigned extraction patterns of their values created manually). After matching a pattern from the dictionary to attribute values of the table, the extracted values are transferred as new values to the corresponding attribute(s) of the new table (manual induction patterns of attribute values, automatic extraction, and manual or automatic data mapping); (4) manual or dictionary-based labelling of table values using attribute names from the new table. Following this, the supervised or semi-supervised methods induce extraction patterns for a given attribute. After the pattern's induction process, patterns are used to match and extract information from the attribute values of the table. The

Figure 13. Induction techniques of wrappers. These techniques are used to extract information from plain text (a text document). Following this, the extracted information is stored in an IS.

extracted values are transferred as new values to the corresponding attribute(s) of the new table (the supervised or semi-supervised induction patterns of attribute values, automatic extraction, and manual or automatic data mapping); (5) the unsupervised extraction of attribute values based on clustering methods. The groups or taxonomies of similar values are created and are then assigned (mapped) as new values of the new table attribute(s).

The manual pattern matching method requires considerable human effort and cannot be scaled easily. The dictionary pattern matching method requires human effort to create the appropriate dictionary. Because of this approach, values can be extracted only from a particular dictionary. The matching step of this method is fast, and the method is scalable. The manual pattern induction solution requires human effort to create appropriate patterns. However, because of this approach, we can extract new values that have not been noted before. A limitation of this method is that we cannot create all patterns to cover all values for large domains. In addition, the supervised and semi-supervised solutions require human effort to create well-labelled data sets for the learning process. In the semi-supervised case, this data set requires lesser data than in the supervised methods. These methods receive more patterns and extract more values than manual techniques. The unsu-

Figure 14. (a) Presents a statistic about the efficiency of hydrants [27]. (b) Presents statistics about hydrant labels [27]. The X-axis contains information on cases, i.e. frequency and cumulative sum. The Y-label presents the categories of cases.

pervised method does not require any human intervention. However, it can very hard to control the quality of results and achieve high precision using this method.

Appendix E: Analysis

E. 1 Use Case: Water Points

Figure 14a, b present additional statistics for *the hydrant* use case.

Figure 14a presents statistics about the efficiency of hydrants. It can be concluded that a majority of the hydrants are efficient (near 73%). In the case of the remaining 25%, nothing can be stated about their efficiency. However, approxi-

mately 2% of hydrants are inefficient. Whereas, Fig. 14b presents statistics of hydrant labels. These labels include information such as hydrant’s identifiers, dates of manufacture, and other relevant parameters. Based on a Pareto chart, it can be concluded that about 78% of hydrants had such labels. The remaining 12% of hydrants do not have such plates (plates were destroyed), or the information was not noted.

Figure 15. Rescue operations carried out during car accidents (the operations frequency > 4%). Figure shows a number of reports and a percentage of rescue operations carried out during car accidents.

Figure 16. A lattice indicates the rescue actions carried out by the rescue units in the car accident sites jointly. Each node contains information about unit names and the number of actions carried out jointly.

E.2 Use Case: Car Accidents

Figures 15, 16, 17, 18, and present additional statistics for the car accidents use case.

Figure 15 shows the frequency diagram of rescue operations carried out during car accidents. The author obtained 15 different and most frequent rescue operations in 205 selected rapports, i.e. there are 205 records and 15 unique names of rescue operations (15 unique values of *operations* attribute, see Fig. 12b). It can be concluded, based on the analysis of Fig. 15, that rescue operations (*frequency* > 4%) most frequently include activities that involve cutting-off electricity and securing an accident site. Another common information that appears in rescue reports is that an accident site was handed over to the police or medical rescue teams. The fire brigades, police, and medical emergency teams usually cooperate in the case of car accidents. Firefighters usually hand over the accident site to the police and assist them until the end of the investigation. They also support the medical teams in rescuing and transporting the injured to hospitals. They frequently conduct other related operations such as towing cars to a safe place, managing traffic, and cutting off gas.

Furthermore, we can obtain information concerning the frequency patterns, i.e. the frequent rescue actions that are performed together. The author extracted the most frequently occurring (*support* $\approx$ 0%, *confidence* $\geq$ 100%) patterns of rescue operations, including (the electricity was cut off, quenching water), (not carrying

Figure 17. Car accidents—location of rescue units involved in the car accident interventions. The red markers indicate the location of the rescue units (Color figure online).

out medical activities, the electricity was cut off, the installation of gas was cut off, securing of accident place), (before the rescue units arrived at accident place the victims had been evacuated, the electricity was cut off, help for an emergency medical services, removal of the wreckages of the vehicles), (management of rescue activities, the electricity was cut off, the installation of gas was cut off, open check and localisation of the fluid leakages, fluid leakage protection, help for the emergency medical services, the accident site has been handed over to the police), (observation of the accident site, securing the accident site, the electricity was cut off, management of traffic, removal of vehicle wreckage, the medical services and police worked in the accident site), and (the assistance, securing the accident site, the electricity was cut off, the medical services and police worked in the accident site, towing a car to a safe place, the police enquiry).

Figure 16 presents an example of a lattice that describes the cooperation of rescue units at car accident sites. By analysing this lattice we can obtain numerous patterns, i.e. we can recognise which rescue units collaborate frequently. The author identified 29 *rafu* and *vfb* rescue units in 205 selected rapports, i.e. we have 205 records and 29 unique names of rescue units (29 unique values of *rescueUnits* attribute, see Fig. 12b). These are the *rafu* and *vfb* rescue units, such as no. 1 (1),

Figure 18 presents an example of a lattice that describes rescue equipment used in car accidents. In this figure, we have several types of acronyms and shortcuts such as the tool *hydra ulic.* (hydraulic equipment), *slrd* (the light car for road rescue), *srd* (the car for road rescue), *slrt* (the light car for technical rescue), *srt* (the car for technical rescue), *slop* (the light operational car), *sop* (the operational car), *srchem* (the car for chemical rescue), *gba* (the firefighting car with barrel and an autopump), *gbam* (the firefighting car with barrel, autopump, and motopump), *gcba* (the heavy firefighting car with a barrel and an autopump), *gcpr* (the heavy firefighting car with the load of firefighting powder), *glm* (the light firefighting car with the motopump).

Analysing the lattice from Fig. 18, we can obtain frequency patterns, i.e. we can recognise what rescue equipment is frequently used in car accidents rescue interventions. The author discovered 29 different rescue equipment types (the equipment concept, see Fig. 8b) in 205 selected rapports. There were 205 records and 29 unique names of rescue equipment (29 unique values of *equipment* attribute, see Fig. 12b). Based on the analysis of the lattice from Fig. 18, we can conclude that the equipment and tools most frequently used in rescue interventions (*frequency* > 4%) are *slrd*, *hydra ulic.* tool, *slrt*, *sorbent* and *gba*. In addition, it can be concluded (*support* $\approx$ 1%, *confidence* $\geq$ 70%) that the following equipment and tools were used at the same time during several particular rescue actions (the orthopaedic board, the tool *hydra ulic.*, the neck brace), (the compact, the *sintan*, the sawdust), (*gba*, *gcba*, *gbam*, *slrt*), (*gba*, *slrd*, the tool *hydra ulic.*, *sop*), (*gba*, *slrt*, the tool *hydra ulic.*, *sop*), (*gcba*, *sop*, *srt*), (*gcpr*, the pump, *sop*, the *sorbent*, *srchem*), (*glm*, *slrt*), (the sand, *slrt*, the tool *hydra ulic.*, the sawdust), (the membrane pump, the neutralizer), (strainer, the *sorbent*), (*slop*, *gba*, the specialised equipment, *srt*), (*sop*, the tool *hydra ulic.*, *gba*), (*sop*, *srd*, the *sorbent*), (the tool *hydra ulic.*, *srt*, *gba*, *sop*), and (the tool *hydra ulic.*, the sawdust, the sand, *slrt*).

Analysis of the patterns indicate that in rescue actions concerning car accidents, hydraulic equipment, pumps, and chemicals are most frequently used. The hydraulic equipment is used to cut and unclench metal parts of the car to free injured occupants trapped in the vehicle. The pumps are usually used to eliminate the leakage of hydraulic fluid such as diesel or maintenance liquids. Furthermore, the chemical agents are used to neutralise such leakages.

References

1. Open source projects related to natural language processing. <http://opennlp.apache.org/>. Accessed 24 July 2019
2. Ordinance of the interior and administration minister of 29 december 1999 on detailed rules for organization of the national rescue and firefighting system. dz.u.99.111.1311 §34 pkt. 5 i 6
3. Gotoh Y, Renals S (2000) Sentence boundary detection in broadcast speech transcripts. In: in Proceedings of ISCA workshop: automatic speech recognition: challenges for the new Millennium ASR-2000, vol 2000, pp 228–235

4. National fire incident reporting system. complete reference guide (2015). https://www.usfa.fema.gov/downloads/pdf/nfirs/nfirs_complete_reference_guide_2015.pdf. Accessed 24 July 2019
5. Resc info insight fights fire with data (2019). <https://opendataincubator.eu/category/resc-info/>. Accessed 24 July 2019
6. Smart data for smarter firefighters (2019). <https://netage.nl/products-2/>. Accessed 24 July 2019
7. Aggarwal CC (2015) Data mining. The textbook, 1st edn. Springer, Berlin. <https://doi.org/10.1007/978-3-319-14142-8>
8. Anderson A, Ezekoye OA (2018) Exploration of nfirs protected populations using geo-coded fire incidents. *Fire Saf J* 95:122–134
9. Bing L (2006) Web data mining: exploring hyperlinks, contents, and usage data (data-centric systems and applications). Springer, New York
10. Boucher T, Yalcin A (2010) Design of industrial information systems. Academic Press, London
11. Brank J, Mladenić D, Grobelnik M (2017) Feature construction in text mining. Springer, Boston, pp. 498–503
12. Cimiano P (2006) Ontology learning and population from text: algorithms, Evaluation and applications. Springer, New York
13. Cimiano P, Hotho A, Staab S (2005) Learning concept hierarchies from text corpora using formal concept analysis. *J Artif Intell Res* 24(1):305–339
14. Formica A (2012) Semantic web search based on rough sets and fuzzy formal concept analysis. *Knowl Based Syst* 26:40–47
15. Haav H (2004) A semi-automatic method to ontology design by using fca. In: University of Ostrava, Department of Computer Science
16. Jayaswal B, Patton P (2011) Design for trustworthy software: tools, techniques, and methodology of developing robust software, 1st edn. Prentice Hall Press, Upper Saddle River
17. Kang YB, Haghighi PD, Burstein F (2016) Taxofinder: a graph-based approach for taxonomy learning. *IEEE Trans Knowl Data Eng* 28(2):524–536. <https://doi.org/10.1109/TKDE.2015.2475759>
18. de Knijff J, Frasnica F, Hogenboom F (2013) Domain taxonomy learning from text: the subsumption method versus hierarchical clustering. *Data Knowl Eng* 83:54–69. <https://doi.org/10.1016/j.datak.2012.10.002>
19. Kreuzthaler M, Schulz S (2015) Detection of sentence boundaries and abbreviations in clinical narratives. *BMC Med Inform Decis Mak* 15(2):1–13. <https://doi.org/10.1186/1472-6947-15-S2-S4>
20. Lareau A, Long B (2018) The art of the after-action review. *Fire Eng* 171(5):61–64
21. Maedche A, Staab S (2000) Mining ontologies from text. In: Proceedings of Knowledge Engineering and Knowledge Management (EKAW 2000), LNAI 1937. Springer
22. Manning CD, Raghavan P, Schütze H (2008) Introduction to information retrieval. Cambridge University Press, New York
23. Miłkowski M, Lipski J (2011) Human Language Technology. Challenges for Computer Science and Linguistics: 4th Language and Technology Conference, LTC 2009, Poznan, Poland, November 6-8, 2009, Revised Selected Papers, chap. Using SRX Standard for Sentence Segmentation, pp. 172–182. Springer, Berlin
24. Mironczuk M (2012) Application of formal concept analysis for information extraction system analysis. *WAT Bull* 61(35):270–293
25. Mironczuk M (2014) Detecting and extracting semantic topics from polish fire service reports. *Appl Math Inf Sci* 8(6):2705–2715

26. Mirończuk M (2014) The method of designing the knowledge database and rules for a text segmentation tool based on formal concept analysis. *Bezpieczeństwo i Technika Pożarnicza / Saf Fire Tech* 34:93–103
27. Mirończuk M (2015) The cascading knowledge discovery in databases process in the information system development. In: 2015 second international conference on computer science, computer engineering, and social media (CSCESM), pp 83–89. <https://doi.org/10.1109/CSCESM.2015.7331873>
28. Mladenović D, Brank J, Grobelnik M (2017) Document classification. Springer, Boston, pp. 372–377
29. Moens M (2006) Information extraction: algorithms and prospects in a retrieval context, Springer international series on information retrieval. Springer, Secaucus
30. Navigli R, Velardi P, Faralli S (2011) A graph-based algorithm for inducing lexical taxonomies from scratch. In: Proceedings of the twenty-second international joint conference on artificial intelligence - volume volume three, IJCAI'11, pp 1872–1877. AAAI Press. <https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-313>
31. Pawlak Z (1981) Information systems theoretical foundations. *Inf Syst* 6(3):205–218. [https://doi.org/10.1016/0306-4379\(81\)90023-5](https://doi.org/10.1016/0306-4379(81)90023-5)
32. Poelmans J, Ignatov DI, Kuznetsov SO, Dedene G (2013) Formal concept analysis in knowledge processing: a survey on applications. *Expert Syst Appl* 40(16):6538–6560
33. Poelmans J, Kuznetsov SO, Ignatov DI, Dedene G (2013) Formal concept analysis in knowledge processing: a survey on models and techniques. *Expert Syst Appl* 40(16):6601–6623
34. Priss U (2006) Formal concept analysis in information science. *Annu Rev Inf Sci Technol* 40(1):521–543. <https://doi.org/10.1002/aris.v40:1>
35. Read J, Dridan R, Oepen S, Solberg LJ (2012) Sentence boundary detection: A long-solved problem? In: 24th international conference on computational linguistics, pp 985–994
36. Sammut C, Webb GI (2011) Encyclopedia of machine learning, 1st edn. Springer, Berlin
37. Sammut C, Webb GI (eds.) (2017) Supervised learning, pp 1213–1214. Springer, Boston
38. Sokolova M, Lapalme G (2009) A systematic analysis of performance measures for classification tasks. *Inf Process Manag* 45(4):427–437
39. Stevenson M, Gaizauskas R (2000) Experiments on sentence boundary detection. In: Proceedings of the sixth conference on applied natural language processing, ANLC '00, pp 84–89. Association for Computational Linguistics, Stroudsburg, PA, USA. <https://doi.org/10.3115/974147.974159>
40. Wolff KE (1994) A first course in Formal Concept Analysis. In: Faulbaum F (ed.) *Stat-Soft '93*. Gustav Fischer Verlag, pp 429–438