Data Mining and Shallow Text Analysis of the Data of the State Fire Service of Poland

K. Kreński, A. Krasuski, and S. Łazowy

Chair of Computer Science, The Main School of Fire Service ul. Słowackiego 52/54, 01-629 Warsaw, Poland

Abstract. There is a vast amount of data created by the State Fire Service of Poland, but there is lack of tools to process that data effectively. The idea of intelligent search tool for natural language sections of EWID database is introduced. Data clustering is performed first and after adding statistical analysis a compact insight into the database is obtained. This allows for creation of controlled vocabulary and then for annotations of actions by the use of bag of words model.

1 Introduction

Each of approx. 500 Fire and Rescue Unit (JRG) of the State Fire Service of Poland (PSP) conducts around 3 fire and rescue actions on daily basis. There is a report created in the internal computer system of PSP named EWID after every single action. The reports comply to the requirements set by [11]. The data collected in EWID database is divided into two sections - structured (database fields) and unstructured (description in natural language (NL)). The structured section contains, among others, the information about the type and size of rescue action, the number of rescue units and the quantity and type of resources used during the action. The total number of database attributes for all above information is 180.

The data described above makes it possible to create statistical information about, e.g., the amount of extinguishing agents used. Of the 180 EWID database several are integer for storing quantities as well as foreign keys of tables containing information description, e.g., the type of the action. Other fields are reals used to store values such as dimensions of action's scene, quantity of extinguishing agents, etc. There are also boolean fields used to mark the usage of particular special equipment and date fields for all time related data. By using the statistics module of EWID - *EWIDSTAT* it is possible to run SQL queries and apply filters to find requested reports. Still, there are no tools to conduct any complex analysis. The NL section contains information which extents the scope of the first (structured) section. The section is made of usually few sentences describing the timeline of the rescue action. Since EWID is a vast source of information about full range of actions, it is reasonable to inspect what useful information can be extracted from it. The authors expect that at present the statistics engine doesn't match the full potential of the database, therefore the research was started on the improvements in this area.

By creating the ontology of EWID content, the reports could be annotated in order to enhance searching in the database. Currently, searching is obstructed by highly inflected forms (the "feature" of Polish language), synonyms, polysemy and lack of context/structure which is typical for texts of this sort. The methodology for ontology creation is to start with inspection of database content with the use of statistical tools, find the interesting terms and place them in the ontology.

CONCURRENCY, SPECIFICATION AND PROGRAMMING
M. Szczuka et al. (eds.): Proceedings of the international workshop CS&P 2011
September 28-30, Pułtusk, Poland, pp. 313-321

2 An Ontology Approach For Searching In EWID Data

Currently, users of EWID perform searching in the system in the structured part only, which has its shortcomings. There is a fixed number of attributes (database columns) which limits the number of sensible queries. On the other hand, the unstructured part of the database, which is filled with the NL descriptions, may contain valuable information which is not present in structured part. Therefore the issue is raised concerning the unstructured part as a subject for user queries. There is an interest in extraction of more detailed information from PSP data [6–10].

Searching the NL documents has its issues. There are language related problems like the necessity to deal with synonyms, homograms, typographic errors and also strictly technical problems like performance and optimization. By adding some structural layer to the NL information, these issues can be to some extent addressed. Ontology is a popular tool which may be applied to solve the structurization problem. The general approach proposed by authors is to create an annotation for every NL section in EWID. The annotation will consist of the leaves of ontology tree and in mathematical sense will be expressed as a multidimensional vector of binary features. In more detail, large dictionary for all EWID data will be created and each word from the dictionary matching the given report will be marked as '1', otherwise '0'. This will result in a long vector of ones and zeros attached to each report, which will constitute an alternative, more machine-oriented representation of the report. Queries will be transformed to vectors in similar manner, which will result in effective searching by just finding the vector(s) which is the most similar to the one that is being searched for.

There are more approaches than just creating ontology to address the problems of text analysis. Ontology creation requires quite amount of human work, but depending on automatic technologies that are based on limited understanding of semantics (such as frequencies of words [3]) can at no means equal handcrafted work of domain experts. Semi-automatic methodologies for creating ontologies also exist [2, 5], but the core of the ontology creation results mostly from experts knowledge.

Once the ontology is created (the details of the approach are described later in this article) and the list of available terms is defined, the next step will be tagging the NL descriptions with the terms from the ontology, like in the example below:

- The Natural Language data:
  After accomplishing the goal, the Fire Rescue Chief together with his Deputy and one of the drivers left the area and the ambulance took the injured driver away.
- Candidate concepts for annotation data:
  Fire Chief, Deputy of Fire Chief, ambulance took away, injured driver
- Concepts that carry no or little meaning:
  After, accomplish, goal, accomplishing goal, fire, ...

The reasons for some concepts being relevant and other irrelevant is that the final goal for the organisation of the data is to make it search-effective. In the example above there are concepts which would constitute a bad query (or bad answer rather) for resulting in too many positives, producing misleading answers. For example, by browsing the reports it is revealed that most of the actions begins with the words "After accomplishing the goal" and querying for such a phrase would result in getting too many positives. In Information Retrieval domain such a situation is named *high recall* and should be avoided. Querying for "fire" should not output the report above for another, obvious reasons.

The proposed methodology to achieve above mechanism would be as follows:

- first there must be vocabulary created enclosing relevant terms of the domain,
- each report will be introduced in database in two separate forms:
  - as it is currently,
  - as annotation for the NL section in the form of *bag of words* [4]
- searching will occur in the annotation database, but the relevant NL section will be the returned answer.

The process of annotating NL sections is as follows: for each term in controlled vocabulary database there is an iteration started which aims at finding the given term in the NL text. The NL processing in the form of declension, lemmatisation, etc. may be applied to the terms prior to the search, in order to find phrases in related, inflected forms: *dowódca akcji, dowódcę akcji, dowódcy akcji, etc.*. Any term (feature) found in the NL text is written as "1" in its relevant position in the vector of features, which is the representation of the annotation. The vector is written to the database as corresponding to the given NL document.

3 The Strategy For Ontology Creation

There is no universal way to create an ontology. Even though some methodologies exist (in the form of general approaches and hints rather than step-by-step procedures), the process of ontology creation is a laborious task and is heavily dependent on the domain experts knowledge and intuition.

Authors' fundamental assumption is that introducing a restricted subset of the NL for all users of EWID system cannot be taken into account. Therefore it must be accepted that there will appear synonyms and inaccurate or misleading constructions in NL sections. Spotting such awkward objects in the texts is, except from organisation of the ontology tree, the main part of this research. In order to complete this task, a semi-automated methods are to be introduced to ease the process of finding how else given concepts may be expressed by the users. The outcome of this task should be the list of the constructions expressing the same concept and this list will be available to the computer system:

the injured was transported to the hospital = the injured was taken = ambulance took away

Then for each list a label must be created, preferably by choosing the most often occurring construction for the given concept, be it "ambulance took away" for the above example. It may be reasonable in some cases to also add comments for less self-explaining lists, which would ease later maintaining. Such lists would address the problem of synonyms - the concepts would be always expressed in the same way.

The hard part is constructing the ontology. The basic question is "which concepts should be placed in ontology?". Some hints may come in the form of a list of questions which are likely to be asked by the system users. The other useful indication of what may have the value is by inspecting the information already introduced to the system. EWID database contains lots of data which creates the risk that simply thinking of all possible questions could not provide their complete list. The alternative is to mine in the data first and then manually decide on what has possible value and find the concepts for the ontology in the process.

4 Statistical Analysis Of Natural Language Sections Of EWID

Some form of evaluation is needed first in order to provide insight to the information that EWID database currently contains. The statistical analysis of the text was performed by the AntConc tool [1]. The following statistics were performed on a sample of data:

```
1. List of single words ordered by frequency:
    116 na
    54  z
    49  i
    48  w
    46  miejsce
    42  po
    ...

2. Collocates (frequent neighbour words for a given word,
   here "miejsce" (a place)):
    46 miejsce
    36 na
    26 zdarzenia
    8  stwierdzono
    4  akcji
    1  zieleni
    ...

3. Concordance (the context of a given word, here "miejsce" (a place)):
    . o przybyciu na miejsce zdarzenia stwi
    . enie oddymiono miejsce zdarzenia prze
    . dojechaniu na miejsce zdarzenia stwi
    . o przybyciu na miejsce akcji przeprow
    . iema w domu na miejsce akcji przybył
    . o przybyciu na miejsce zdarzenia ugas
    . ...

4. N-grams, here most frequent 3-grams:
    18        na miejsce zdarzenia
    13        działania polegały na
    12        po przybyciu na
    12        przybyciu na miejsce
    11        polegały na zabezpieczeniu
    8        dojeździe na miejsce
    ...
```

Listed above are just the most frequent objects from the corpus, which doesn't mean the most useful. The *Recall* for these entities is high, which means too many documents would be returned if the query included these terms. On the other end of the list there would be entities which are infrequent. Often, the reason for infrequency of some phrases comes from the fact that they are not typical concepts for the corpus and are candidates for being dropped. The more of such infrequent objects are placed in the ontology, the higher will be the *Precision*, which has a tendency to output too few answers for each query. The desired set of terms to be included in the ontology would be something in the middle of these two extremes. Finding the upper and lower limits of this "middle"

is the task for the experts after they can view the data in an approachable shape. The example below expresses the idea for the ordered list of 3-grams:

```
na miejsce zdarzenia     -
działania polegały na    -   High Recall (too frequent concepts)
przybyciu na miejsce     -


      ...                +
      ...                +   The middle (interesting concepts)
      ...                +


śmigłowca do karetki     -
że prowadząca Beata      -   High Precision (too infrequent concepts)
Łosicach jeden pas       -
```

The process of collecting the controlled vocabulary is semi-automatic (humans analyses with computers support). The human input allows for more careful organization of the data, by putting them to their respective parts of the hierarchy and naming all the nodes of the tree. Having the data well organised will allow not only for advantages in later searching, but should also improve the data organisation process itself. Once the concepts are collected in the hierarchy (or probably multiple hierarchies), there may be more *relations* added than just *is-a* relation resulting from position in the tree, e.g. between a fireman and a fire there may be a relation "extinguish". Authors expect that useful relations candidates will be revealed once the controlled vocabulary is completed. The relations have their advantage that they may be defined universally and don't have to be explicitly named for every single annotation. Except from the relations there may be *restrictions* defined, e.g. *a high-rise building is not a low-rise building*. The usefulness of the above ontology features will be evaluated after the vocabulary is complete and the final outcome may affect the way of annotation and searching process. At present, the best candidate for expressing the annotation are the multidimensional vectors of binary features ("0" for each absent feature and "1" for each present feature).

Such vectors need to be sorted if they are to be subjects to comparisons, but the details of how to sort them are not of authors concern at this stage, as this doesn't influence current work. At a rough glance it is expected that there must be intervals created which would contain terms from the same parents (in the ontology tree). These intervals would be ranges inside the vectors, e.g. bits 1-5 are the information about the building, bits 6-20 relate to the proceedings taken and so on. The ranges would be taken under account when calculating the similarity among the vectors. The definitions of the ranges (lower/upper limit) would be kept outside the vector in form of indices. The resulting vectors will be multidimensional (the degree of dimensions will equal the size of the vocabulary) and one of the existing algorithms for operating on long vectors will be applied.

5 Pre-Processing Prior To The Shallow Text Analysis

Choosing the candidates for ontology concepts may turn out a difficult undertaking when working with full database of unstructured texts. Since there is a corresponding, structured information for each report, this structured data was used for clustering and narrowing the scope of fire and rescue actions. The stage of pre-processing the data in order to shape them for shallow text analysis was

performed with the standard Data Mining (DM) tools. The classical DM methods applied to the structured part of EWID database provided some important and useful information which could be used for separate analyses, but in this case the goal of the data mining was the facilitation and support of text analyses. The goal of the clustering was to create the distinguishing sets of fire and rescue operations. This would then ease ontology creation by allowing for work with just one set at a time instead of the full text corpus.

EWID is composed of two distinct parts: a) the structured part of predefined database attributes (information table) and b) the natural language (NL) part. The approach was to have both parts clustered separately and then have logical AND applied to the clusters in order to increase clustering quality. Unfortunately, clustering of NL part was not successful and only the structured part was used to define the final clusters.

The data were first preprocessed by standard cleaning and fixing procedures. The structured part was standardized. In order to improve quality of clustering, the NL part was lemmatized. In computational linguistics, lemmatization is a process of determining the lemma for a given word, i.e. all the inflexed forms are flattened to the basic form. This is particularly the issue in Polish language, which is very rich in inflexed forms. Lemmatization was performed in Morfologik, an open source Polish morphological analyzer based on ispell dictionary [12].

The dataset was first divided into three groups: fire incidents, local threats and false alarms. Then clustering was performed only on the fire incidents and the other two were excluded.

Clustering is an automatic process which requires that the target number of resulting clusters is defined in advance and the initial clustering aimed at finding this target number. PAM [16] was chosen as the clustering algorithm and a sample of 10000 incidents was evaluated. The accuracy of resulting clusters was determined using Silhouette width [18] and Calinski Harabasz index [13].

The number of clusters in the experiment was varied from 50 to 700 and the $S(i)$ and $CH(k)$ were calculated. Figure 1 depicts the result of the experiment for Silhouette width.

The Silhouette width grows with the increase of the number of clusters. Around the number of 300 clusters, the $S(i)$ stabilizes at the level 0.1 and doesn't significantly increase anymore. The reason is that one-object-clusters start to appear ($S(i)$ for this case is equal 0). Therefore, it was reasonable to set the target number of clusters at the value of 300, which is the minimum value where $S(i)$ reaches 0.1.

Silhouette width can vary from -1, which means very poor quality of clusters, to 1 denoting very good quality. Therefore the value 0.1 achieved in the experiment is considerably low. In the next step, full Silhouette analysis for $k = 300$ clusters was performed to check for the reason of the low index value. Figure 2 depicts the result of the experiment.

According to the figure 2 there is a set of clusters which have a fairly high value of $S(i)$ and the set of clusters for which the value of $s(i)$ is below 0. The minus values represent the incidents which rarely occur and are significantly different from other cases. They weaken the $S(i)$ because there are no similar incidents in the cluster. Therefore, the further analyses were performed on clusters where $s(i) > 0.3$.

After determining the target number of clusters, the clustering process was performed on full database. In order to handle large size of database the CLARA [15] clustering method was used.

The next stage was clustering the NL part of EWID using algorithm called *Latent Semantic Analysis* (LSA) [14, 17]. The basic idea of LSA is to first create concepts for the given text corpus and then assign each single word from a document to a concept. The result is that documents can be expressed in Latent Semantic Space which a) is considerably compact and b) allows for finding indirect similarities between documents.

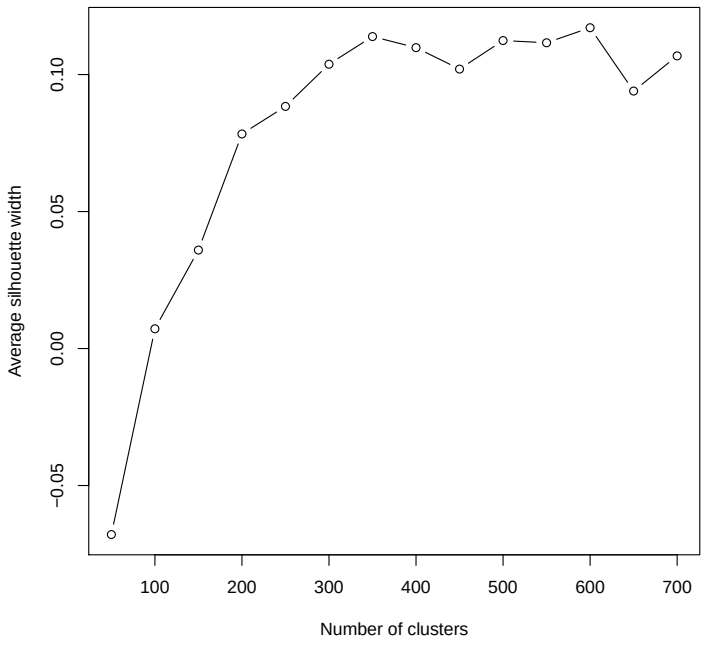

Fig. 1. Silhouette width against the number of clusters (high values are better)

However, there were difficulties in clustering the NL part as the computer resources were not sufficient to conduct the LSA clustering, which turned out to be very demanding for computer power. The clustering of NL part was not completed and the final clusters resulted from just the structured part of the database.

6 Conclusions And Future Work

Improvements to searching in natural language sections of EWID database is possible by introducing the annotations. The terms to be inserted to the actual bags of words must be first collected in the form of an ontology/dictionary. In order to create the ontology, the compact insight to the data is needed. After the data is clustered and subjected to the statistical analysis it is the time for the experts to consider which terms are of the importance and collect them in the ontology.

The next step will be gathering the group of experts and actual creating of the ontology. It is likely that only some part of the ontology (based on just one or few of the clustered sets) will be created, as ontology building is a laborious task and needs reasonable time and human resources. The resulting, limited ontology may be sufficient to validate whether the obtained searching mechanism is successful. In case the result is reasonable, the research and the obtained ontology may be used as guidelines to ease the completion of the missing parts of the ontology.

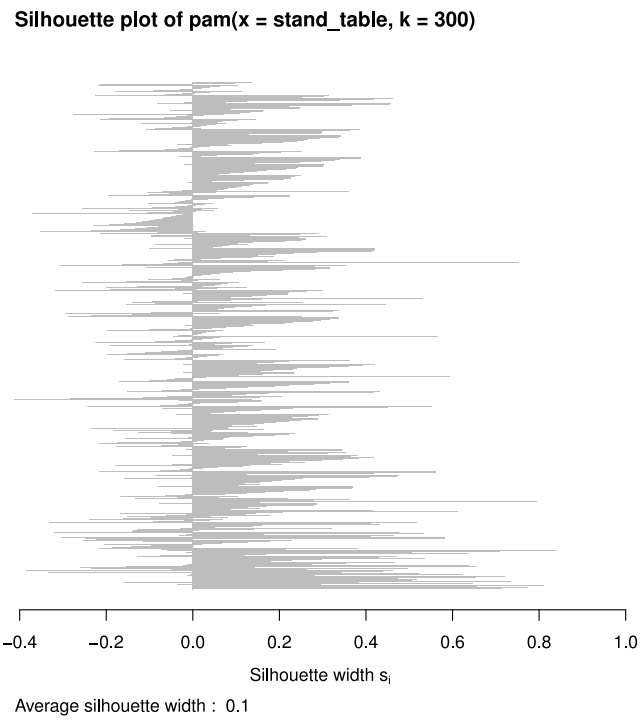

Fig. 2. Silhouette width for k=300

References

1. Anthony, L.: AntConc: A learner and classroom friendly, multi-platform corpus analysis toolkit. In: Proceedings of IWLeL 2004: An Interactive Workshop on Language e-Learning. pp. 7–13 (2005)
2. Jaimes, A., Smith, J.: Semi-automatic, data-driven construction of multimedia ontologies. In: 2003 International Conference on Multimedia and Expo, 2003. ICME'03. Proceedings (2003)
3. Joachims, T.: Text categorization with support vector machines: Learning with many relevant features. Machine Learning: ECML-98 pp. 137–142 (1998)
4. Jurafsky, D., Martin, J., Kehler, A., Vander Linden, K., Ward, N.: Speech and language processing. Prentice Hall New York (2000)
5. Kietz, J., Maedche, A., Volz, R.: A method for semi-automatic ontology acquisition from a corporate intranet. In: Workshop: Ontologies and text. Citeseer (2000)
6. Krasuski, A., Maciak, T.: Rozproszone bazy danych, możliwości ich wykorzystania w Państwowej Straży Pożarnej. Zeszyty Naukowe SGSP 34, 23–42 (2006)
7. Krasuski, A., Maciak, T.: System wspomagania decyzji w Państwowej Straży Pożarnej. Wykorzystanie rozproszonych baz danych oraz metody wnioskowania na podstawie przypadków. Zeszyty Naukowe SGSP 36, 23–42 (2008)
8. Krasuski, A., Maciak, T., Kreński, K.: Decision Support System for Fire Service based on Distributed Database and Case-based Reasoning. Studies in logic grammar and rethoric 17(30), 1–11 (2009)
9. Kreński, K., Maciak, T., Krasuski, A.: An overview of markup languages and appropriateness of XML for description of fire and rescue analyses . Zeszyty Naukowe SGSP 37, 27–39 (2008)

10. Mirończuk, M., Kreński, K.: Koncepcja systemu ekspertowego do wspomagania decyzji w Państwowej Straży Pożarnej. Inżynieria wiedzy i systemy ekspertowe red. Adam Grzech [et al.] (2009)
11. Rozporządzenie Ministra Spraw Wewnętrznych i Administracji z dnia 29 grudnia 1999 r. w sprawie szczegółowych zasad organizacji Krajowego Systemu Ratowniczo-Gaśniczego. Dz. U. z dnia 31 grudnia 1999 r.
12. Morfologik – About the project. `http://morfologik.blogspot.com/2006/05/about-project.html`
13. Caliński, T., Harabasz, J.: A dendrite method for cluster analysis. Communications in Statistics-Theory and Methods 3(1), 1–27 (1974)
14. Deerwester, S., Dumais, S., Furnas, G., Landauer, T., Harshman, R.: Indexing by latent semantic analysis. Journal of the American society for information science 41(6), 391–407 (1990)
15. Kaufman, L., Rousseeuw, P., Corporation, E.: Finding groups in data: an introduction to cluster analysis, vol. 39. Wiley Online Library (1990)
16. Van der Laan, M., Pollard, K., Bryan, J.: A new partitioning around medoids algorithm. Journal of Statistical Computation and Simulation 73(8), 575–584 (2003)
17. Landauer, T., Foltz, P., Laham, D.: An introduction to latent semantic analysis. Discourse processes 25(2), 259–284 (1998)
18. Rousseeuw, P.: Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. Journal of computational and applied mathematics 20, 53–65 (1987)