TEXT MINING AND KNOWLEDGE DISCOVERY WITH ONTOGEN 2.0

Mladen Tomaško
Police Academy, Police College and Jožef Stefan International postgraduate school
e-mail: mladen.tomasko@gmail.com

ABSTRACT

This paper presents a text mining technique used for extracting knowledge from two databases containing the same set of documents. The first database contains only abstracts, while the second one contains full text documents. Data preprocessing and processing is described, followed by a description of problems and encountered doubts. Special attention was given to the comparison of results taken from these databases. Some remarks on the tool used for data mining were given and for a conclusion some future researches in this field were proposed.

1 INTRODUCTION

In the last decade or two, a large amount of data was collected all over the word. Handling it is almost an impossible task. When we search for some specific topic, we are lost because we find numerous irrelevant documents ("data deluge").

Text mining technique can help us to extract important data more efficiently. This tool sorts our documents by searching similar words or phrases in documents. It also gives us a possibility to interfere on search process and to actively improve accuracy. Text mining is a relatively "young" technique; it is a part of data mining or – in wider context - a part of data analysis. The roots of this tool lie in researching learning computer to understand ontology, documents concept and implementation of this for document classification, document clustering and document visualization. It is extremely useful for web crawling and extracting information we are interested in.

There are numerous tools in the market for text mining, some specialized for certain fields (e. g. medicine, law etc.) and some more general. The main point of good text mining tools is concept understanding and most efforts today are dedicated to improve that property.

Spending less time searching for relevant information gives us more time for researching and working. A short overview how data mining can help us is explained in this article through a practical example.

2 TEXT MINING

2.1 Data description

For practical work we had two different text files, properly organized and prepared for deeper analysis. Preparing and collecting data can take a while and can require some special skills, but it is an important step to obtain good final results.
We had two datasets; one with only short description (abstract) of documents and the other with full text. These two files give us a good base for explaining how efficient text mining will be. Since we are searching for new knowledge in these files, we will use different methods to make our work fast and accurate.
There are 4571 documents about chemistry. Each document is represented with title and short abstract (in the first file) or with title and full text (in the second file). The first file has about 670 Kb and the larger one has about 12,5 Mb. For the real world these are relatively small databases, but representative enough that the result could be generalized.

2.2 Data preparation

Data were taken from different resources, mainly from the web. The use of different characters coding tables brought about inconsistencies in the final file. The first task was to eliminate these characters, which follows the wrong documents classification.
Then we tried to start with data mining process, but there was another type of noise. In numerous documents there were messages from web pages about the incompatibility between browser and frames these web sites use.
To avoid this problem we prepared a set of words and add it to the stop word list. The result was much better concept suggestions.

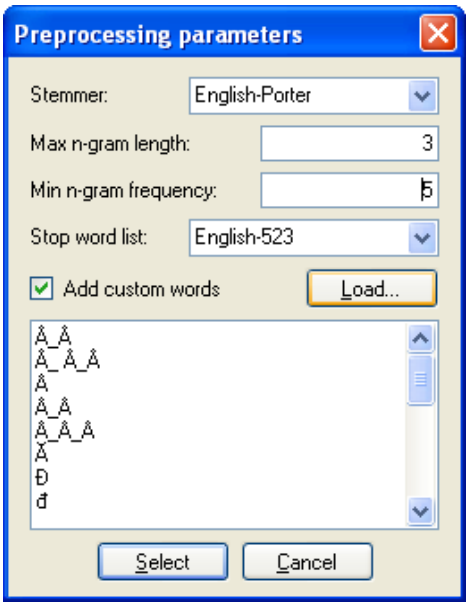
2.3 Tool used for text mining

For text mining we used OntoGen 2.0 tool [1, 3], that is a semiautomatic tool which allows us ontology and concept visualization and a lot of ways to improve concept definitions. Despite some initial difficulties described in

details later, the overall user experience using OntoGen was fair.

2.4 Working with data

In Figure 1 we can see a window used to make some general adjustments before importing data. We have the possibility to add our own stop words and to give some other parameter when importing data.
This function for adding custom stop words is very handy when we have data with lot of specific noise. In our example, different character coding produced noise and we added this characters to an existing stop word list.



With data imported, the programme extracts a selection of keywords. In the beginning, it is good to visualize our concept, to have a picture how documents are organized on the basis of extracted keywords.

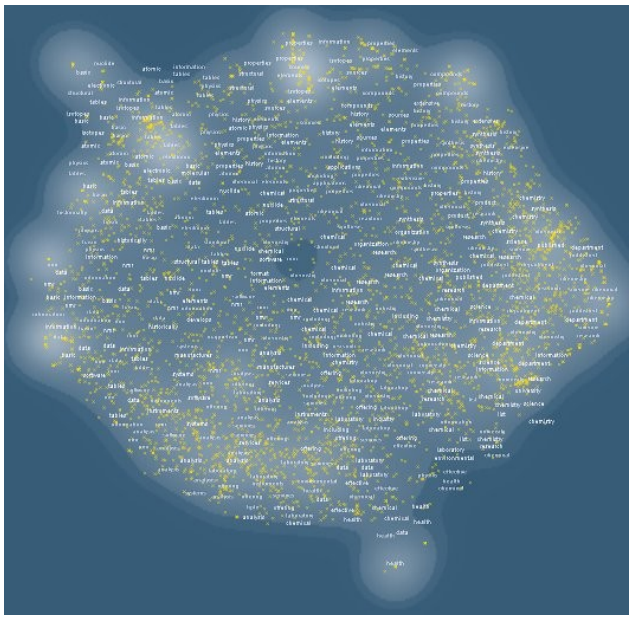


Figure 2: Concept visualization (file with abstracts)

Figure 2 shows how our data were arranged. In this database, there are only short abstracts, usually a sentence or two, and this is obvious in the way how documents are organized in the graph. Documents are arranged in a circle and it is hard to determine how many suggestions to choose. The next step was not clearly seen from this visualization. Fortunately, we can experiment with different numbers of suggestions and see which one gives best results.
There are two (potentially three) visible areas with superior density. This could be a good starting point. If we compare results given by other file (with full text), we can see noticeable improvement in quality.
Figure 3 shows results obtained from second (larger) file. It is clear that it is much easier to divide data on basis of the second picture.
Therefore, logical decision will be split it into three areas with good similarity. But after numerous try and check

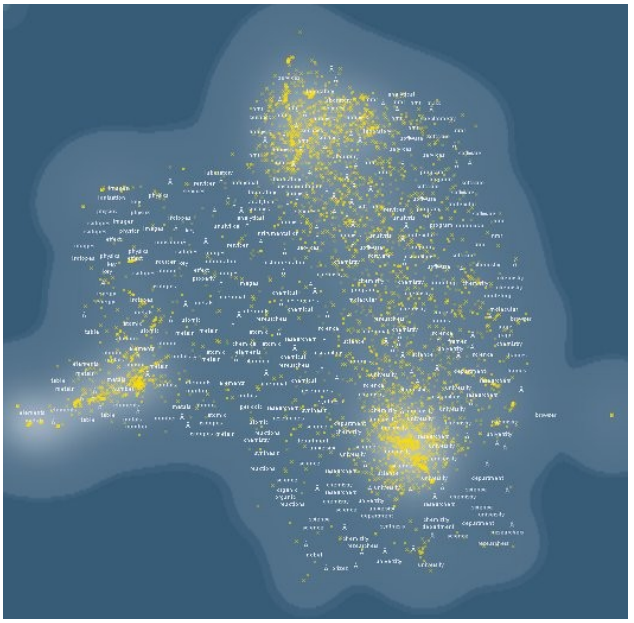


Figure 3: Concept visualization (file with full text)

attempts we found that dividing into five branches give best results. Once again we could determine that better data means better final results. Despite of more text and potentially more noise, this data are much easier to manage. Contrary to our expectation we can easy and fast obtain better results with this database than with previous one containing only abstracts. But there is also the dark side of the moon. More data means more calculations. And in this case difference is easy to measure. First data were calculated about 44 seconds vs. 55 second with full text documents. The time needed for visualization is not linearly connected with the size of data. However, there is a noticeable delay when importing larger database.
In our example both databases are small. The larger one is almost that easy for compute as the smaller one. With big databases this will be a problem and we do not have enough information to accept right decision with which database to

go further. One another thing is also obvious. In larger database documents are just in size of normal abstract and not like ordinary articles or documents which are usually at least four or five pages long. This can follow us in trap to take a wrong conclusion because of insufficient information.

After first steps taxonomy trees look like on the two figures below. The first one is much better defined due to sufficient text in the documents and data mining programme has enough information to do a good classification during the first step.

With appropriate methods we can improve similarities and accuracy in our concept.

2.5 Improving accuracy and similarity of concept

In OntoGen we can work in supervised and unsupervised way. Supervised means that we exactly know how our data must be organized. We used unsupervised way, but we tried to improve accuracy (similarity) in certain branches. We can join branches together, divide it into more sub-branches or just exclude some documents that are not directly connected with our concept. Another possibility is using sim graph in Concept's documents tab to include or exclude documents.

Using all this we finished with a relatively small taxonomy tree with branches that can be divided into more sub-branches if necessary.

Figure 4 shows the final tree is shown. Branches have good similarity and after random checking and evaluation we realized that accuracy is good enough. Working this way is time consuming (because of user interaction), but in the end we get far better results than results given by programme itself.

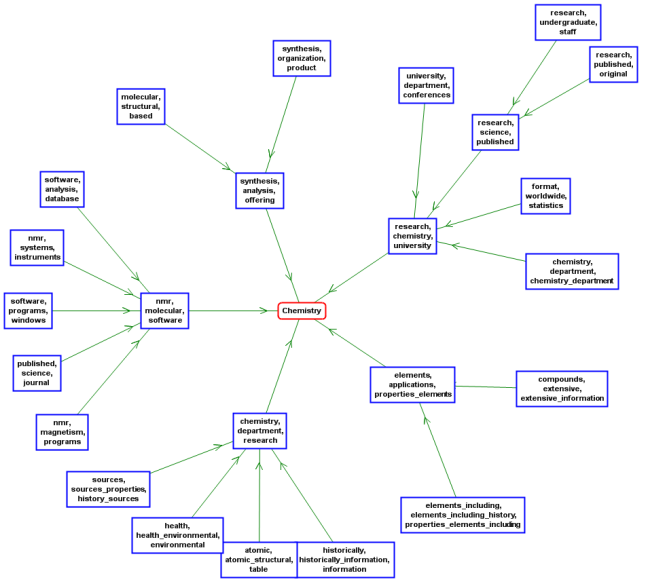


Figure 4: Ontology visualization (abstracts)

If we compare this tree with the next one in the Figure 5. we can see they are very similar despite the fact that the first one is produced from abstracts and the second one from full text documents.

As mentioned before, the main difference is the amount of time needed to obtain similar results. With full text documents with no intensive human work we can get almost the same results. However, for large databases it is important that we can obtain such abstract from documents. That can save us a lot of computing time and simplify building of a taxonomy tree.

If we need a model with high accuracy, help of an expert is needed. He can help us to improve our ontology concept manually. In our example with reasonably small database, it was easier to experiment with data. On large databases a proper strategy is needed which saves time and money.

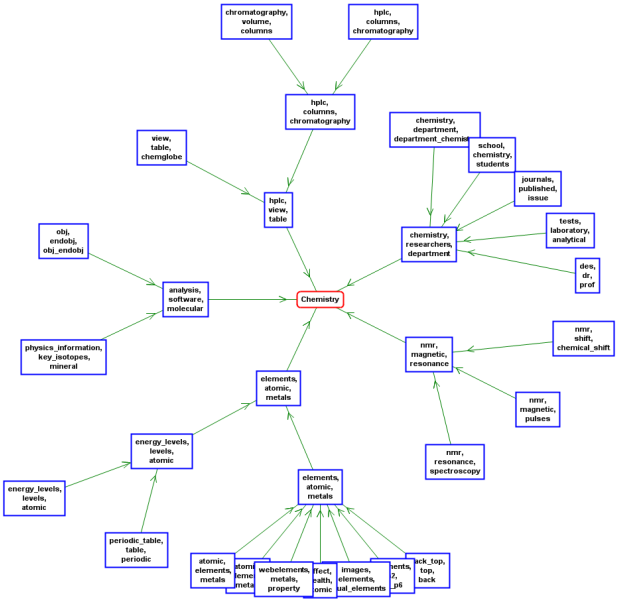


Figure 5: Ontology visualization (full text)

The tree in Figure 5 shows the selected final ontology. Accuracy is good enough in almost all branches, whole process to obtain this result was straightforward and no special knowledge about chemistry was needed. There is enough difference between branches and similarity inside branch is good.

We didn't change names what programme proposed, even thought calculation of SVM keywords gives slightly different results. We noticed that if user manually change the name of node then all words are in the same line and there is no visible way how change it. That follows in unclear graph and we decided to stay with programme's proposed names.

The hardest part of building tree is proper division of nodes who have relatively low similarity. This part usually demands lot of manual work.

As an example how visualization of concept with good similarity looks, there is one of node from our taxonomy tree in Figure 6.

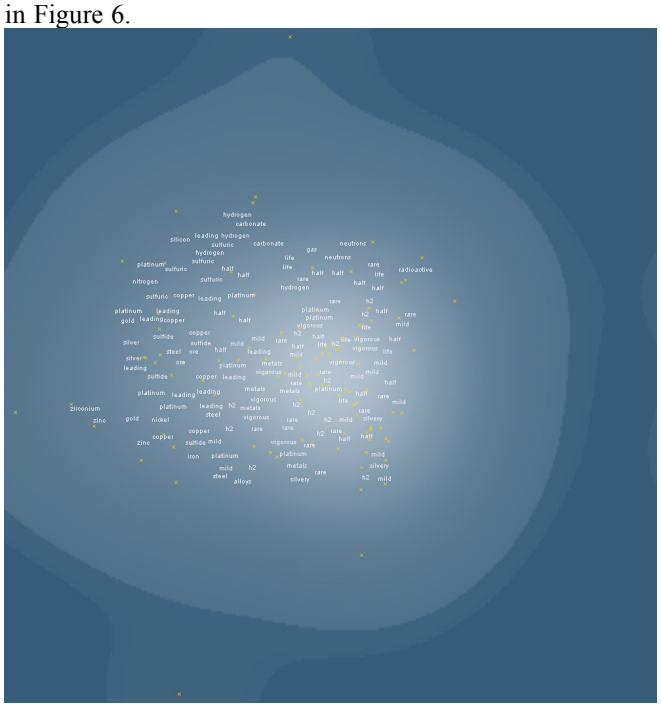


Figure 6: Concept visualization with good similarity

4 REMARKS ON ONTOGEN 2.0

When using a programme for a while, I have noticed some problems. When trying to visualize some concept, usually on

Figure 7: OntoGen non fatal error message

2nd or 3rd level, the system stopped responding and we had the following error. I unsuccessfully tried to locate this file on my computer. After this error I was able to save my work and opened it again. Clicking on OK followed OntoGen to close.

Another error also occurred during the concept visualization phase. This one was a bit more fatal. OntoGen stopped working, there was no possibility to save the data and the only solution was to close OntoGen. The error occasionally occur in this phase of visualization (see Figure 8).

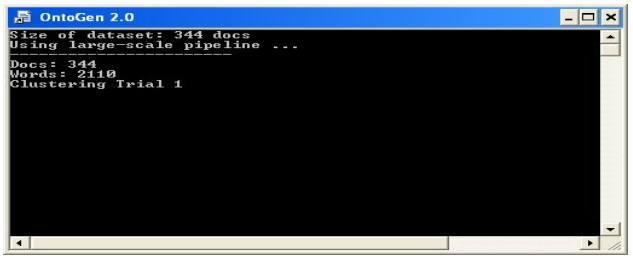


Figure 8: OntoGen fatal error

3 CONCLUSION

When trying to build accurate data models, it is very important that we have some knowledge on data or there is an expert available. Results are then easier to evaluate and we can classify data closer to user needs.

Next important point is properly defined stop words list. This helps removing noise from data and follows to higher programme accuracy. It is hard to generalize results obtained in this short research, but OntoGen gives us enough power to compete with everyday's classifying demands.

Working with OntoGen was also an opportunity to deeper understand the way how to use text mining methods in relation to real world problems. Nowadays collecting data is an easy task. Using it or classify it without right tool is rather difficult. Billions bytes of data are collected on our computers every day. Large logs (we usually deleted this without looking into it) are unmanageable. Text mining gives us a chance to extract important information and make our work more efficient.

Text mining can be also used for cleaning databases of garbage data. For instance, in domain of police a significant amount of data is false classified (manually) by mistake of a policemen or administration staff. Text mining can help to recognize such situations and classify them correctly.

The real power of text mining is visible when researching intensive fields. Terabytes of data were collected through the years and some important discoveries are still hidden somewhere in it. Digitalization of books and archives gives text mining a new challenge to compare and discover new knowledge.

References

[1] Fortuna, B., Grobelnik, M., Mladenić, D, **Semi-automatic data-driven ontology construction system.** *HCI International 2007*, July 2007, Beijing.

[2] Drew, R., **Text mining tools take on unstructured data**,http://www.computerworld.com/databasetopics/businessintelligence/story/0,10801,93968,00.html, accessed 23. 1. 2009

[3] B. Fortuna, M. Grobelnik, D. Mladenic. **Semi-automatic Data-driven Ontology Construction System.** *Proceedings of the 9th International multi-conference Information Society IS-2006, Ljubljana, Slovenia*.

[4] http://kt.ijs.si/Dunja/textgarden/, accessed 23. 1. 2009

[5] http://www.nactem.ac.uk/resources.php, accessed 23.1. 2009

[6] Redfearn, J., Text Mining http://jisc.ac.uk/publications/publications/bptextminingv2.aspx, accessed 23.1. 2009