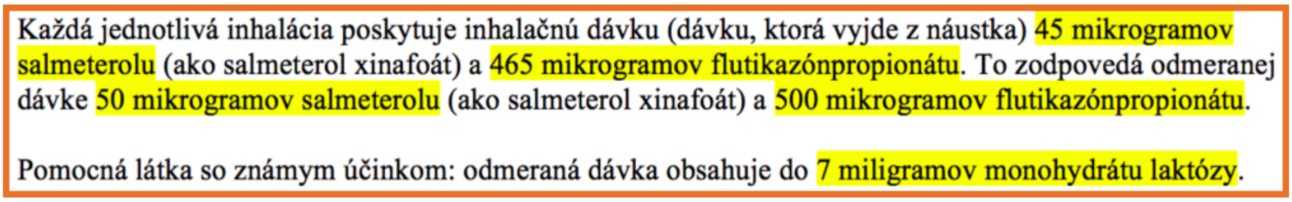
BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY

Excel@FIT 2016
http://excel.fit.vutbr.cz

Information Extraction from Loosely Structured Text

Matej Minárik*



Každá jednotlivá inhalácia poskytuje inhalačnú dávku (dávku, ktorá vyjde z náustka) 45 mikrogramov salmeterolu (ako salmeterol xinafoát) a 465 mikrogramov flutikazónpropionátu. To zodpovedá odmeranej dávke 50 mikrogramov salmeterolu (ako salmeterol xinafoát) a 500 mikrogramov flutikazónpropionátu.

Pomocná látka so známym účinkom: odmeraná dávka obsahuje do 7 miligramov monohydrátu laktózy.

Abstract

We are experiencing a data explosion. There is an undoubtedly increasing amount of data published every day on the internet. Scientific articles, journal papers, books, images, movies, music, tweets, statuses and a whole lot more. With this data explosion, new problems arise. Where and how to store these data and how to efficiently search for relevant data? This work should provide a semi-automatic solution to extract relevant data from medicine information sheets. These data should serve as a source data for search engine in the future. Information sheets are stored as semi-structured documents on the website of National institute of Medicine control of Slovak republic.

We propose a semi-automatic method, which is able to extract active substances and basic information about symptoms, which suggest usage of specific medicine. Our method have some features of supervised and some features of unsupervised information extraction. Our solution was experimentally evaluated and achieved $\sim 70\%$ accuracy. However, we are focusing on a subset of all relevant information included in these documents.

Our solution is inspired by previous unsupervised information extraction, but we have implemented a novel method which includes transferring natural language into intermediate representation with help of Slovak national corpus[1] and information extraction from this representation based on manually created rules.

Keywords: information extraction — data mining — medicine information sheets

Supplementary Material: *N/A*

*xminar29@stud.fit.vutbr.cz, *Faculty of Information Technology, Brno University of Technology*

1. Introduction

Some people say, that nowadays we live in an era of data. People, machines, sensors and even gadgets we carry in our pockets generate large amount of data every day. Companies like Facebook, Google and Apple are building large data centers all around the world in order to deal with these data. This work is a part of a search engine, which should be able to provide relevant information about medicines. There is a state institute in Slovak republic, which is responsible for medicine registration and approval. This institute publishes documents on its website about all medicines available in Slovak republic.

These documents are published as Microsoft Word documents in semi-structured form. In order to build a proper search engine, we need to extract some relevant pieces and provide them in structured form. This project focuses on creating a semi-automatic information extraction software. This software should be able to extract relevant information from these documents and store them in structured form.

There are several information extraction tools available. Most of these tools originated as a contribution to MUCs conferences held during 1990s. These conferences had form of a competition, where several teams were trying to construct an information extraction system. One of such systems is CLAWS[2]. Almost all of these tools solve standard NLP (nature language processing) problems, such as tokenization (process of dividing text to elementary units called tokens), lemmatization (process of finding the dictionary form of specified word or token) and named entity recognition (subtask of information extraction to classify elements in predefined categories, such as names of persons, organizations, etc.). Unfortunately, all of these tools are either trained and developed for specific domain, e.g. extracting relevant information from resume[3], or they are focused on texts in English[4].

Our solution is able to extract basic information about medicine. We are able to extract a short paragraph containing symptoms, which serve as an indicator to use this medicine. Besides that, we are able to extract active substances included in specific medicine. There are several producers of medicines all around the world. There are cases, when two separate producers have medicine with the same active substance, but different supplementary substances. In such cases, we want to provide the information, that there is an alternative medicine, which can be used in case of specific symptoms.

We have a solution which is able to convert Microsoft Word documents to a plain text form with some additional data about document styles. From this text we can extract short paragraph containing symptoms and active substances. We can find $\sim 70\%$ of active substances included in analyzed documents.

2. Theoretical background

Information extraction can be solved with several methods. One of them is classification. Classification is a process of assigning a class to object (record) based on its attributes. An example of such attribute may be a record from relational database of students. A specific student may have some grades from courses he graduated and we want to classify this student to either ”pass” or ”fail” class, which corresponds to passing or failing the final school exam respectively. Classification is a process divided into two parts: a) training part, b) testing part. During the training part, we are trying to create the internal model, based on which we will be able to classify these students. During the testing part, we are evaluating accuracy of trained model. There are several models which are used to solve classification tasks: decision trees, Neural networks, Naive Bayes classifier and others. Detailed description of these methods is out of scope of this paper.

Classification has some downsides. The most relevant and obvious downside is, that in order to train a model, we need a large amount of training classified data. We, in our domain do not have this kind of data. In order to create training corpus, a domain expert is needed. This task is not just time consuming, but a substantial time would be needed in order to find such expert.

There are other methods, which do not need large classified corpus. These methods are based on a ”seed” of data that we want to classify. For example, if we want to extract active substances and their corresponding medicines, we can provide a few substance-medicine pairs as an input. This method is frequently referred to as bootstrapping[5]. DIPRE[6] method is based on a simple duality premise: if we provide this method with quality seed data, then we can find quality appearances, if we can find quality appearances, then we can extract new patterns based on which we will extract new quality appearances. DIPRE method can be summarized into few algorithmic steps:

1. start with a small set of quality seed data
2. find all appearances of these data and extract the context, in which they appeared
3. generate new patterns from appearances, this step needs to be carefully implemented, because too general patterns will extract many wrong appearances, but too strict pattern won't be able to find all appearances
4. find new appearances based on generated patterns
5. repeat until satisfied

Context of extracted appearances is following:

$$(medicine, substance, order, url, prefix, middle, suffix) \tag{1}$$

1. medicine - name of medicine
2. substance - name of active substance in corresponding medicine
3. order - true of false, whether the medicine was first or the substance was first
4. url - path to specific document with appearance
5. prefix - several tokens before appearance

6. middle - tokens between author and title
7. suffix - several tokens after appearance

This method was tested with 5 initial substance-medicine pairs. After 5 iterations, there was 15 257 unique substances with corresponding medicine out of which just a few substances were wrong.

3. Active substances extraction method

Our solution is inspired with DIPRE method, discussed in previous section. In order to achieve better results, we use Slovak national corpus[1], which is available at [http://korpus.juls.savba.sk]. Analysis of these documents showed, that all of them are separated into paragraphs and these paragraphs have similar headings and order numbers. Therefore, we can very easily parse the document and determine based on these headings and their order numbers what paragraph are we actually processing. Based on these assumptions, in order to extract active substances, we will find the correct paragraph with appearances of active substances.

First of all, we need to convert MS Word documents, which are either in OLE 2 (.doc) format, or OOXML (.docx) format. We achieve this by using Apache Tika^{TM}[1] library which is able to work with different formats. In order to benefit from document structure we convert these documents to HTML format. Converted HTML documents are composed of paragraphs and headings. Some headings are also converted to paragraphs. We also use Jsoup[2] library, which is capable of scrapping, parsing and manipulating HTML documents. With Jsoup library, we can very easily extract all paragraphs in document and iterate over all of them.

3.1 Tagging

Slovak national corpus provides different information about words. We are extracting tags, which include:

1. part of speech
2. paradigm
3. gender (masculine, feminine, ..)
4. singular/plural
5. case (Nominative, ...)

Each of these parts corresponds to one letter in obtained tag. For example: ”S - noun, S - substantive, f - feminine, s - singular, 1 - Nominative and the tag will be SSfs1”.

Almost each active substance is next to some kind of quantity expression, e.g. ”...10 ml of water”. We have analyzed these occurrences and came up with a very simple set of regular expressions, which are able to effectively locate and identify these quantity expressions. In general, we are focusing on locating a number and then we try to locate one of several possible units, e.g. ml, mg, mmol, umol, etc.

Our method is converting the text into internal representation and then, based on manually created rules, extracting active substances. The internal representation for ”... contains 10 ml of water ...” can be ”SSfs2 quantity quantity short SSfs4”. ”SSfs2” and ”SSfs4” are tags obtained from Slovak national corpus. ”quantity” tags are obtained from our very simple quantity classifier, which is based on regular expressions. The basic overview of our conversion method:

```
for each document in documents
  for each paragraph in paragraphs
    for each token in paragraph
      if token is quantity
        return "quantity"
      else
        return snr.token
    end
  end
end
```

3.2 Extraction rules

Then we go through all converted tokens and manually extract ”rules” from converted text. Rules represent tags, which corresponds to active substance appearances in documents. Some examples of rules are:

- ”SSfs1”, ”SSfs4”, ”Quantity”, ”Quantity”
- ”Quantity”, ”Quanity”, ”ASfx4”

these rules can for example refer to these occurences:

- ”monohydrát”, ”laktózy”, ”10”, ”ml”
- ”1000”, ”mg”, ”paracetamolu”

Then, we iterate over all documents, convert its contents to internal representation and look for appearances of these rules. Each appearance of such rule we mark as active substance appearance.

Basic information about medicine and symptoms, which indicate that one should start using that concrete medicine can be extracted very easily by finding the correct paragraph and extracting all text inside that paragraph.

4. Experimental evaluation

At this time, there is no reference database of active substances of medicines. In order to evaluate accuracy of our solution, we have analyzed and manually

[1] https://tika.apache.org
[2] http://jsoup.org

extracted active substances from 30 documents. This work resulted in a reference file, which includes 137 extracted active substances from those 30 documents. We passed these documents to our solution and then we examined output from our program with reference file.

Table 1. Experimental results

Description	Result
Totally correct	91
Partially correct	17
False positives	21
True negatives	8
Total	137

In table 1, we can see the experimental results. 91 active substances were correctly and fully extracted. 17 substances were extracted partially. We considered for partial extractions all extractions, which had some additional extracted words, or some words were missing. These extractions can be valuable in most cases, however, there were just 4 extractions, which were different from reference ones just by one short word. 21 extractions were words, which did not include active substances. These extractions were some other words, which had an quantity information near them and the words were in the exact same form as was some of the active substances. These extractions refer to generality of our rules. However, in the context of a search engine, these extractions are not so dangerous. Most of these can be removed in post-processing, or we can develop an internal dictionary which will serve as a reference register of active substances in order to remove false positives. This register can be manually refined after each group of parsed documents. Moreover, we assume that people using our search engine would ask the engine which other medicines, have the exact same active substances as the one they use. In this kind of queries, false positives are not much of a problem. Last, but not least, there were 8 true negatives. These extractions were missing in our output file, but were present in reference file. Most of the missing extractions did not have rules of their form in our program. Extending training corpus of documents would solve this issue, however there is a subtle question of the size of training corpus, which would be sufficient enough.

In order to evaluate our solution and based on previous assumptions, we evaluate the precision like this:

$$\frac{91+4}{137} * 100 = 69.34\% \tag{2}$$

91 fully correct extractions with 4 almost fully correct extractions are almost 70% out of all reference active substances.

5. Conclusions

We stressed the importance of all information extraction and data mining works and contributions. We examined supervised and unsupervised information extraction methods, took the best from both approaches and implemented our solution in order to extract active substances and basic information about symptoms from documents available about medicines in Slovak republic.

We have created a solution which is capable of extracting information from medicine information sheets available online.

We evaluated our solution on 137 reference active substances, which were manually extracted from 30 reference documents. Our solution reached $\sim$ 70% precision according to our tests and reference file created.

There is much space for future work. There are other relevant data in these documents, such as how and when one should take concrete medicine, what are the contradictions, what medicines you must not use with other and so on. Besides that, original idea was to create a search engine and this work should serve just as a tool to get relevant data. I am planning to extend this solution with a dictionary, which will be created during the extraction process and will serve as a reference register. This dictionary should be refined after each group of documents by domain expert.

Acknowledgements

I would like to thank my supervisor Ing. Radek Burget, Ph.D. for his valuable reviews of my work and help. I also would like to thank my girlfriend for her patience while I was writing this paper.

References

[1] Slovenský národný korpus. prim-6.0-public-all. *Bratislava: Jazykovedný ústav Ľ. Štúra SAV 2013. Dostupný z WWW: http://korpus.juls.savba.sk*, 2013.

[2] Roger Garside. The claws word-tagging system. 1987.

[3] Kun Yu, Gang Guan, and Ming Zhou. Resume information extraction with cascaded hybrid model. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 499–506. Association for Computational Linguistics, 2005.

[4] Michael Schmitz, Robert Bart, Stephen Soderland, Oren Etzioni, et al. Open language learning for information extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 523–534. Association for Computational Linguistics, 2012.

[5] Roman Yangarber. Scenario customization for information extraction. Technical report, DTIC Document, 2001.

[6] Sergey Brin. Extracting patterns and relations from the world wide web. In *The World Wide Web and Databases*, pages 172–183. Springer, 1998.