End-to-end Acronym Expander System for Bulgarian language

Thesis Design for MSc IS

Iliya Georgiev
University of Amsterdam
Amsterdam, The Netherlands
iliya.georgiev@student.uva.nl

ABSTRACT

Scientists from different fields such as biology, chemistry, and physics are using acronyms to make their communication faster. The use of acronyms saves time and space, but it makes the text more complex. Not everyone is an expert and has knowledge in these fields, therefore a system for acronym expansion is needed. This system would identify acronyms and find their expansion, so one can understand the meaning of the acronym with ease. An acronym expansion (AE) can be found in the text in a predefined database or can be extracted from the internet. This thesis will focus on the AE problem within the Bulgarian language. The report will provide an extension to an existing end-to-end system for AE to support the Bulgarian language and add support for acronym construction without using parentheses. Moreover, a new data set will be developed and it will be used to test the system. Finally, the author will consider the case where multilingual acronyms are used within the text. The system will be evaluated by comparing a baseline, the current system - end-to-end system for acronym expansion, and the new system based on the precision, recall, F1-score of the new data set, and the execution time it takes to process it.

Main supervior UVA: João Lebre Magalhães Pereira
(j.p.pereira@uva.nl)

1 INTRODUCTION

In recent years, the number of acronyms used in both the scientific and nonscientific domain has drastically increased [1]. Acronyms (or abbreviations) are a short form of longer phrases, usually constructed from the first letter of each word in the phrase. The reasons why people are using acronyms are mainly to reduce the time needed to express themselves and space while writing news articles, scientific papers, and posts on social media.

Whenever people are reading through a document and there is an acronym used in it, they immediately look for the meaning of this acronym. Usually, after the first appearance of the acronym in the text, it is common that the expansion follows immediately after. However, when those acronyms are generally accepted in the field, often their expansion is omitted [1]. This usage of abbreviations raises the problem of acronym expansion (AE). If the acronym expansion is not available in the text, one should understand the context of the text to determine which expansion the author intended to use within the text. Experts in different fields are introducing abbreviations that are applicable for their domain, however, those are not always globally unique. For example, the acronym 'DM' means 'Diabetes Mellitus' in the medical domain and 'Direct Message' in the technical domain.

The above-mentioned problem can be solved by building a system that considers the following two steps: **Acronym Identification (AI)** - find the acronym and its expansion in the text and **Acronym Disambiguation (AD)** - find the correct expansion that corresponds to the acronym. Over the past two decades, there has been significant research towards finding a solution for those two steps [7, 10–12, 15]. The limitation of these studies is that they focus only on Latin-based languages, such as English [10, 12]. Since the Bulgarian language (BL) is using the Cyrillic alphabet and not Latin, it is difficult for such systems to perform AI and AD effectively. A natural processing language model that accepts the BL is needed for the system to support Bulgarian documents more effectively[4].

This thesis will aim to contribute to an existing end-to-end system for acronym expansion. The system currently operates in English, and it is in the process of development by the supervisor of this project (João Lebre Magalhães Pereira). The system takes a document as an input and it outputs a document with all of its acronyms expanded. The system works best using MadDog-in [15] for AI and Support Vector Machine (SVM) [5] and Doc2Vec [8] for AD. This thesis will add the Bulgarian language to this system. This improvement will include expanding the system to support the Cyrillic alphabet as well as adding Bulgarian data set that will be used to train the algorithm used by the system. This will allow future researchers in the field of acronym expansion to use the system for all Cyrillic-based languages such as Ukrainian, Russian, and Serbian. Moreover, new AI techniques will be implemented in the system. Apart from identifying an expansion followed by an acronym inside of the parentheses, they will also identify other patterns that do not include parentheses, such as those based on hyphens ("-"). These patterns are using the following construction EXPANSION - ACRONYM or ACRONYM - EXPANSION. These techniques will be language-independent and therefore they should work with both English and Bulgarian. The functionality of the current system will be used as a baseline to evaluate the performance of the features mentioned above. To accomplish this, the following research question is formulated:

What is the impact on the recall, precision, and F1-measure of the expanded end-to-end system including techniques for AI specifically designed for the Bulgarian language when using the Bulgarian data set that will be developed?

In order to support the main research question, the following subquestions are formulated:

- *How well is the system performing in terms of speed, recall, precision, and F-1 measure when it uses the newly introduce*

no-parentheses technique compared to the existing end-to-end system?

- *To what extent the new no-parentheses technique influences the speed of the system when the Bulgarian language is used for acronym expansion compared to when it operates in English?*
- *How well is the system performing in terms of speed, recall, precision, and F-1 measure when the acronyms in the text consist of more than one language (e.g. English and Bulgarian)?*

The remainder of this thesis proposal is structured as follows. Section 2 focuses on state-of-art research that is related to the acronym expansion. Section 3 describes the methodology that will be used in this thesis. Section 4 assesses the risk there is in this project, while Section 5 discusses the project plan.

2 RELATED WORK

This section focuses on existing research in the field of acronym identification in Section 2.1 and acronym disambiguation in Section 2.2. Moreover, it introduces previous work on the Bulgarian language in the world of Natural language processing (NLP) in Section 2.3.

2.1 Acronym identification

Acronym identification is considered when the expansion of the acronym is available in the text. A considerable amount of literature has been published on this topic showing the two main approaches using rule-based and machine learning (ML) algorithms [7, 10, 12, 14]. Most of the studies are using the notion of precision and recall to evaluate their proposed algorithms [7, 10, 12, 14]. Precision is the number of correctly retrieved acronym-expansion pairs divided by the total number of retrieved acronym-expansion pairs. Whereas the recall is the number of correctly retrieved acronym-expansion pairs divided by the total number of acronym-expansion pairs in the data set [10].

Pustejovsky et al. [10] proposed two solutions to the AI problem using regular expressions(90% precision and 63% recall) and syntactic constraints (99% precision and 61.9% recall). The approach of syntactic constraints was used on a pre-processed data set annotated with additional syntactic information. In this way, the authors were tagging strings and phrases, which allowed them to highly constrain the context where they would have to look for acronym expansion. Moreover, the paper introduced the Medstract Gold Standard Evaluation corpus used for the evaluation of acronym expanders in the medical domain. However, the algorithm focuses on identifying acronyms by looking only at the left-hand context, such as ACRONYM (EXPANSION). The study did not consider right-hand context, where the acronym is defined first and the expansion comes later. For example, PIN - Personal Identification Number is right-hand context, while Personal Identification Number - PIN is left-hand context. This had a critical impact on the recall level resulting in only about two-thirds.

Furthermore, Schwartz et al. [12], introduced an algorithm that can detect acronyms using both left-hand and right-hand context, if parentheses are used to define those. The algorithm creates a set of pairs acronym and expansions and uses iterations to check which is the most accurate pair. The proposed solution achieved 95% precision and 82% recall on the MEDLINE data set - 1000 MEDLINE abstracts randomly selected by using the search query for the term "yeast".

Sohn et al. [14] proposed an automated evaluation of the accuracy of the acronym that selects the most probable pair of acronym and expansion. The authors have defined seventeen different strategies of how an acronym can be matched with the expansion. The proposed system achieved 97% precision and 85% recall on the Medstract Gold corpus.

Kuo et al. [7] used machine learning to achieve 95.86% precision at 84.64% recall on the AB3P corpus. The authors of the paper use the feature extraction technique to construct a feature vector consisting of 4 distinct feature sets. These features are used to train four different ML algorithms - Support Vector Machine (SVM), Naive Bayes, Logistic Regression, and Monte-Carlo Sampling Logistic Regression. Moreover, the authors introduced the BIOADI corpus containing 1200 abstracts from the medical domain, on which their algorithm achieved 93.52% precision at 79.95% recall.

Veyseh et al. [15] proposed an acronym identification model inspired by Schwartz [12]. The study expands the algorithm by adding more rules to it. The Acronym Detector rule considers all of the words with at least 60% of their letters are upper-cased and the number of characters is between 2 and 10 to be an acronym. The authors also add a rule where they consider only capitalize words to be part of the algorithm that looks for algorithms. Moreover, a rule which accounts for a match between the characters in the acronym and the words in the expansion was introduced. The algorithm achieved 89.98% precision at 87.56% recall for identifying acronyms and 96.45% precision at 79.53% recall for identifying acronym expansions. These results were based on the SciAI benchmark data set [9].

2.2 Acronym Disambiguation

Acronym Disambiguation is needed when the acronym expansion is not available in the text and the acronym has more than one possible expansion. In this case, the system will need to determine the right expansion according to the context.

Charbonnier et al. [3] proposed an unsupervised way to disambiguate acronyms. The research uses a word embedding approach to the AD problem. The authors use the full text of the documents in their database for the algorithm after cleaning the data from NLTK stopword list words, words that are less than 2 characters or they occur less than 5 times in the corpus. The algorithm achieves 84% accuracy on small contexts and 86% accuracy on larger contexts from the NOA Corpus.

MadDog [15] is a system that requires two resources, a dictionary that holds an acronym and expansion and a model that can find the right expansion based on the context of the data. They have obtained the dictionary by exploiting their AI rule-based model described in Section 2.1 on different domains (Wikipedia, Arxiv papers, Reddit submission, Medline abstracts, and PMC OA subset). The authors created a dictionary that contains 426 389 unique acronyms and 3 781 739 unique expansions. A supervised model was trained on the Massive Acronym Disambiguation (MAD) data set to predict the correct expanded form of the acronym. The system achieved 92.27% precision at 85.01% recall on the SciAD data set.

2.3 NLP methods for the Bulgarian language

This section describes different methods for processing the Bulgarian language.

Simov et al. [13] introduced the BulTreeBank Project, in which the main goal was to prepare the Bulgarian language for automatic processing by creating a syntactically annotated data set using morphosyntactic tagging, named entity recognition, part-of-speech disambiguator, and partial parsing. To create the national corpus, the authors created three subsets of data - a core set of sentences, a treebank, and a text corpus. The core set of sentences contains more than 2500 sentences extracted from grammar books, where the treebank refers to syntactically annotated sentences with a total word count of around one million. The text corpus gathers a wide document collection from multiple domains. It is multi-layered linguistic annotated and its size is over a hundred million words.

Kapukaranov et al. [6] performed sentiment analysis for movie reviews in Bulgarian. They used textual features such as words, emoticons, n-grams, and lexicon of movie reviews for sentiment analysis. In addition to those the study also used contextual features such as movie length, country, genres, actors, director, average user rating, IMBD score, and Cinexio score. The paper focuses on adding a new data set for movies reviews in Bulgarian containing more than 10000 reviews. The study used three different techniques of processing the annotated data set:

- Classification - SVM with linear kernel
- Regression - same SVM tool, features, and parameters as the classification technique but with predicted numerical value - support vector regression
- Ordinal regression - tries to fit the data into regions but at the same time predicts the values and position in the space.

Out of the three algorithms, the regression worked the best on this data set with a combination of contextual and textual features.

Cherecharov et al. [4] proposed a web-based system for teaching the Bulgarian language. The system introduced an NLP module for the Bulgarian language that can perform a variety of tasks. These include automatic morphological analysis and synthesis, verification of syntactic agreement, automatically placing stress, and automatic processing of complex verb forms. This allows users to search for all forms of a word within the text. Moreover, the user can replace the word with different forms or with completely different words if they are within the same part of speech.

3 METHODOLOGY

This section will focus on the methodology that will be used in this thesis. Data collection and Data preparation are introduced in Sections 3.1 and 3.2, where the implementation of the system is discussed in Section 3.3. Section 3.4 focuses on the evaluation of the results from the research.

3.1 Data collection

The data will be collected from Wikipedia - Bulgaria[1]. The system will be focused on a generic domain and therefore the extraction of articles will be done within 10 generic categories - Psychology, BioMedical, Sports, Politics, Financial, Engineering, Computer science, Math, Education, and Biology. The author will manually extract at least 20 documents per topic and will save these documents as *.txt* files on the server. The articles will be randomly selected without taking any requirements into account. These documents will be later annotated by the author and external readers.

3.2 Data preparation

To be able to use the gathered data for this research, it needs to be pre-processed. This process includes text pre-processing of the Wikipedia data and merging multiple documents to a data set. A custom annotation system will be built by the author and other participators in the project. This system will allow external readers and the author to annotate the collected data. This will be done by providing the user with an extracted text and asking him/her to match all acronyms within the text with the right expansion if there is one. In case the expansion is not found in the text, the user will have to annotate the acronym and its expansion according to the context of the text. After the annotation is performed, then the text of the annotated document with the annotated acronyms and their expansions will be stored on the server. This collection will be then merged and converted to a data set used to test the system. The data will be distributed into 3 different data sets to be used for the three different parts of the system - Acronym Identification, Acronym Disambiguation, and Acronym Expansion. The AI data set will compose of more articles within text expansion, while the AD data set will mostly use articles where the expansion is not inside the text. For the AE, all of the articles will be used for a test data set.

3.3 Implementation

Currently, the end-to-end system recognizes expansions of the acronyms, only if the acronyms are in parentheses. For example, the pair Personal Identification Number (PIN) will be recognized, however, the pair, Personal Identification Number - PIN will not be recognized. The addition to the current implementation will be to add support for the Bulgarian language and add acronym recognition when the acronym is not in parentheses in the first encounter. At the moment, the system is divided into two parts to effectively solve the acronym expansion problem.

For AI, two strategies will be tested and observed which one will perform better in the Bulgarian language. The first one will be using a rule-based approach, implementing the Schwartz et al. [12] algorithm and the MadDog version of this algorithm [15]. The second strategy used for the implementation of the Bulgarian language will be the machine learning approach using SciBERT models[2]. Moreover, to recognize the acronym when no parentheses are used new AI strategy will be added. This strategy will recognize other patterns for acronyms that do not include parentheses, such as an expansion followed by an acronym inside the parentheses ("-"). These patterns will identify acronyms and expansions that use the following structure, EXPANSION - ACRONYM or ACRONYM - EXPANSION. For example, the current AI techniques - MadDog-In [15] and Schwartz [12] used in the existing system will not recognize the following acronym-expansion pair, Natural language processing

[1]https://bg.wikipedia.org/wiki/%D0%9D%D0%B0%D1%87%D0%B0%D0%BB%D0%BD%D0%B0_%D1%81%D1%82%D1%80%D0%B0%D0%BD%D0%B8%D1%86%D0%B0

- NLP, because the acronym is not within parentheses - Natural language processing (NLP). To recognize acronyms and expansions outside of the parenthesis the author will implement and test the rule-based AI techniques proposed by Yarygina et al. [16] using regular expressions and pattern matching approach introduced by Mohammed et al. [11]. This new strategy will not be dependent on the language and therefore it will be available for all languages that are implemented in the system.

For the AD, the system has to look up the acronym in an external database containing documents and the pairs of acronyms and expansions that can be found in these documents. After this, the system decides based on the context which expansion fits best. To do this, the author will try different techniques such as:

- Representator - this technique will use Term Frequency-Inverse Document Frequency (TF-IDF) to calculate the frequency of the acronym usage in the documents, Latent Dirichlet Allocation (LDA) to determine the topic of the document, and Doc2Vec to discover how similar the documents are.
- Classification - approaching the problem with ML, these techniques will include SVM, Logistic Regression (LR), and Random Forests (RF).
- Combination - combining different techniques might increase the accuracy of the system. This is why the author will use a different combination of techniques and evaluate which one fits best for the Bulgarian language.

3.4 Evaluation of the results

To evaluate the results of this study, the author will separate the system into three parts - acronym identification, acronym disambiguation, and acronym expansion. As a base benchmark, this study will use a random approach, where the disambiguation of the acronym is randomly selected from the possible list of candidates. Moreover, the system which is already available in English will also be used as a benchmark, comparing the techniques discussed in section 3.3. The evaluation will be using the newly created data sets in Bulgarian to assess the performance of the system when used in Bulgarian. The author will use the following performance metrics:

- *Precision* - number of correctly retrieved acronym-expansion pairs divided by the total number of retrieved acronym-expansion pairs.
- *Recall* - number of correctly retrieved acronym-expansion pairs divided by the total number of acronym-expansion pairs in the data set.
- *F1-score* - harmonic mean of the precision and recall.

$$F_1 = 2 \cdot \left(\frac{precision \cdot recall}{precision + recall} \right)$$

- *Execution time* - related to the amount of time needed for the system to process the data set and not the amount of time needed to train the system

These metrics will be used to evaluate the performance of the system when using previously annotated data sets.

Moreover, as mentioned earlier the thesis will focus on handling multilingual disambiguation, or the use of more than one language when defining acronyms in the document. To evaluate what is the impact of this, the author will look into the measures mentioned above when multilingual acronyms are used. It will be investigated if the system performance is worse when the text contains multilingual acronym expansion pairs, for example, "Интерфейс за малки компютърни системи (SCSI)".

4 RISK ASSESSMENT

The main risk in this research is the resources, hence the manpower to construct a proper data set in the Bulgarian language. If there are not enough people who could help with the annotation, then the possibility of not extensive and rich Bulgarian data set rises. This could have an impact on the results of this study and it can alter the evaluation process. A possible solution for this could be that the author will contact universities in Bulgaria and ask for support from the linguistic departments.

Another risk related to this research is the use of English acronyms in the Bulgarian language. In most cases, when using a term that is generally accepted for a specific domain, the authors are using the Bulgarian name and the English acronym for this term. For example, "Интерфейс за малки компютърни системи (SCSI)" is used in a Wikipedia article related to hard drives [2]. Possible solution for this issue will be to alter the AI technique to match the number of words in the expansion with the number of symbols in the acronym even though there is no match between the initial letter of the words with the acronym symbol. Using this approach, the system will be able to identify acronyms without being dependent on the acronym and expansion to be in the same language.

5 PROJECT PLAN

The duration of this thesis project will be 4 months. Table 1 shows the project planning and deliverables in more detail. During the first month of the project, the author will work on the data collection and data preparation and set up the current system locally. Then in the second month and third month, the author will implement the different models and techniques to support the expansion of Bulgarian acronyms described in Section 3.3. Furthermore, a sufficient amount of work in the writing of the thesis will be made. In the fourth month, the plan is to evaluate the results and try to improve the models. Moreover, the finalized thesis will be delivered along with a thesis presentation that will be used to defend the thesis.

REFERENCES

[1] Adrian Barnett and Zoe Doubleday. 2020. The growth of acronyms in the scientific literature. *eLife* 9 (2020), 1–10. https://doi.org/10.7554/eLife.60080

[2] Iz Beltagy, Kyle Lo, and Arman Cohan. 2020. SCIBERT: A pretrained language model for scientific text. *EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference* (2020), 3615–3620. https://doi.org/10.18653/v1/d19-1371 arXiv:1903.10676

[3] Jean Charbonnier and Christian Wartena. 2018. Using word embeddings for unsupervised acronym disambiguation. *COLING 2018 - 27th International Conference on Computational Linguistics, Proceedings* (2018), 2610–2619.

[4] Stoyan Cherecharov, Hristo Krushkov, and Mariana Krushkova. 2017. Nlp Module for Bulgarian Text Processing. *CBU International Conference Proceedings* 5 (2017), 1113–1117. https://doi.org/10.12955/cbup.v5.1080

[5] Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine Learning* 20, 3 (sep 1995), 273–297. https://doi.org/10.1007/BF00994018

[2] https://bg.wikipedia.org/wiki/%D0%A2%D0%B2%D1%8A%D1%80%D0%B4_%D0%B4%D0%B8%D1%81%D0%BA

Week	Dates	Actions | Deliverable
10-13	7 th March - 3rd April	Data exploration Data collection - Wikipedia articles
14	4 th April - 10th April	Further investigation of works related to this research
15	11th April - 17th April	Introduction finalized Data collection finalized
16	18th April - 24th April	Related works section completed Data Preparation
17	25th April - 1st May	Setup existing system Data preparation finalized
18	2nd May - 8th May	Models creation for the Bulgarian language
19-20	9th May - 22th May	Implement parentheses feature, Methodology section updated
21	23rd May - 29th May	Test the system with the created data set, feedback from the results
22	30th May - 5th June	Evaluate results, Methodology section finalized, Results section updated
23	6th June - 12th June	Results section finalized
24	13th June - 19th June	Discussion, Conclusion and Abstract finalized
25	20th June - 26th June	Finalizing thesis and preparing thesis defense presentation
26	27th June - 3rd July	Defense week

Table 1: Project planning

[6] Borislav Kapukaranov and Preslav Nakov. 2015. Fine-grained sentiment analysis for movie reviews in Bulgarian. *International Conference Recent Advances in Natural Language Processing, RANLP* 2015-Janua (2015), 266–274.

[7] Cheng Ju Kuo, Maurice H.T. Ling, Kuan Ting Lin, and Chun Nan Hsu. 2009. BIOADI: A machine learning approach to identifying abbreviations and definitions in biological literature. *BMC Bioinformatics* 10, SUPPL. 15 (2009), 1–10. https://doi.org/10.1186/1471-2105-10-S15-S7

[8] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations ofwords and phrases and their compositionality. *Advances in Neural Information Processing Systems* (2013), 4178–4179. arXiv:1310.4546

[9] Amir Pouran Ben Veyseh, Franck Dernoncourt, Quan Hung Tran, and Thien Huu Nguyen. 2021. What Does This Acronym Mean? Introducing a New Dataset for Acronym Identification and Disambiguation. (2021), 3285–3301. https://doi.org/10.18653/v1/2020.coling-main.292 arXiv:2010.14678

[10] James Pustejovsky, José Castaño, Brent Cochran, Maciej Kotecki, and Michael Morrell. 2001. Automatic extraction of acronym-meaning Pairs from MEDLINE databases. , 371–375 pages. https://doi.org/10.3233/978-1-60750-928-8-371

[11] N. Saneesh Mohammed and K. A.Abdul Nazeer. 2013. An improved method for extracting acronym-definition pairs from biomedical Literature. *2013 International Conference on Control Communication and Computing, ICCC 2013* Iccc (2013), 194–197. https://doi.org/10.1109/ICCC.2013.6731649

[12] Ariel S. Schwartz and Marti A. Hearst. 2003. A simple algorithm for identifying abbreviation definitions in biomedical text. *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing* (2003), 451–462. https://doi.org/10.1142/9789812776303_0042

[13] Kiril Simov, Petya Osenova, Milena Slavcheva, Sia Kolkovska, Elisaveta Balabanova, Dimitar Doikoff, Krassimira Ivanova, Alexander Simov, and Milen Kouylekov. 2002. Building a linguistically interpreted corpus of Bulgarian: The bul tree bank. *Proceedings of the 3rd International Conference on Language Resources and Evaluation, LREC 2002* (2002), 1729–1736.

[14] Sunghwan Sohn, Donald C. Comeau, Won Kim, and John W. Wilbur. 2008. Abbreviation definition identification based on automatic precision estimates. *BMC Bioinformatics* 9 (2008), 1–10. https://doi.org/10.1186/1471-2105-9-402

[15] Amir Pouran Ben Veyseh, Franck Dernoncourt, Walter Chang, and Thien Huu Nguyen. 2021. MadDog: A web-based system for acronym identification and disambiguation. *EACL 2021 - 16th Conference of the European Chapter of the Association for Computational Linguistics, Proceedings of the System Demonstrations* (2021), 160–167. https://doi.org/10.18653/v1/2021.eacl-demos.20 arXiv:2101.09893