pHealth 2021
B. Blobel and M. Giacomini (Eds.)

doi:10.3233/SHTI210615

Automatic Extraction and Decryption of Abbreviations from Domain-Specific Texts

Michil EGOROV [a,1] and Anastasia FUNKNER [a,2]
[a] *ITMO University, Saint Petersburg, Russia*

Abstract. This paper explores the problems of extraction and decryption of abbreviations from domain-specific texts in Russian. The main focus are unstructured electronic medical records which pose specific preprocessing problems. The major challenge is that there is no uniform way to write medical histories. The aim of the paper is to generalize the way of decrypting abbreviations from any variant of text. A dataset of nearly three million medical records was collected. A classifier model was trained in order to extract and decrypt abbreviations. After testing the proposed method with 224,307 records, the model showed an F1 score of 93.7% on a valid dataset.

Keywords. Clinical text, medical records, natural language processing, abbreviations

1. Introduction

Electronic health records (EHR) are widely used to build models for predicting the process of healthcare provision [1]. Such texts contain terms, specific abbreviations, and acronyms, whose decryption depends on the field of usage, particular medical institution, or even particular specialist [2]. These factors make research of the task of extraction complicated. This paper presents a method of automatic detection and decryption of abbreviations.

Recognition of well-established abbreviations and acronyms is usually carried out with the help of dictionaries [3] and marked data [4], and mainly addresses data in the English language. MeDAL [3] contains medical texts with abbreviations and their possible decryptions. Models pre-trained on this dataset improve their metrics by 0.2-2%.

For the Russian language, as one of the low-resource languages, this area is not well researched. The author of [5] considers the problem of extraction and decryption of abbreviations from the Corpus of Legislative Acts of the Russian Federation. The paper considers approaches to topic modeling of texts to identify words that are similar in terms of use and contexts. As the author highlighted in their work, automatic decryption of abbreviations using such approaches is not accurate enough.

[1] Corresponding author. e-mail: egorovmichil9@gmail.com.
[2] Corresponding author. e-mail: funkner.anastasia@gmail.com.

2. Methods

This section sequentially describes the steps taken to extract and decrypt abbreviations.

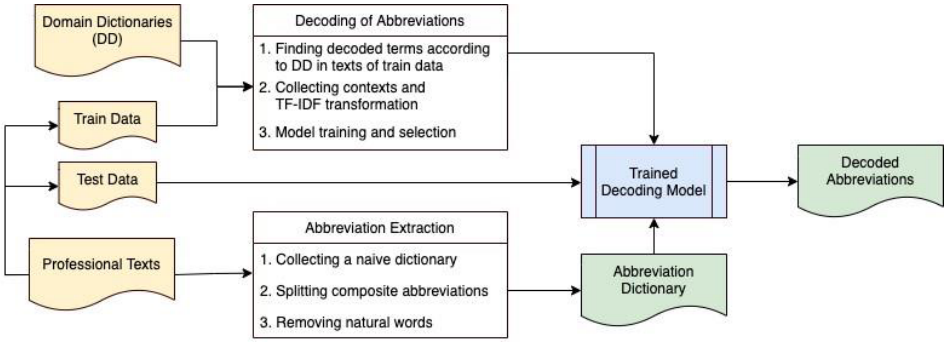

Figure 1. Diagram of methods for extraction and decryption of abbreviations.

Firstly, a dictionary of abbreviations needs to be collected (Fig. 1). Keep in mind the assumption that a word is an abbreviation, if at least one half of its letters are upper-case. This allows to compile a naive abbreviation dictionary. However, this dictionary can include composite abbreviations, for example "BP.HR". Such abbreviations may be distinguished by punctuation. Moreover, the naive dictionary also contains natural words that were written in upper case. As an abbreviation often contains unnatural character pairs, the average entropy of character pairs in a word allows to remove natural words from the dictionary using the Shannon entropy formula. This means the more entropy, the more natural the word is. It remains to choose the threshold value of entropy.

The algorithm has two stages: training and applying. For the training stage, one should make a corpus with encoded abbreviations. Words in the corpus need to be normalized and filtered using a list of stop words. In the stage of applying, the trained model is used with an abbreviation and its context as input and a decoded term as output.

In order to train the decryption model, the input text is divided into three parts: context before and after an abbreviation, and its characters. Contexts are collected with a window and transformed into two Term Frequency-Inverse Document Frequency (TF-IDF) vectors. The stack of bag-of-characters vectors and two transformed TF-IDF contexts become features. Decryption models can only be trained on two TF-IDF vectorizer, since using bag-of-characters vectors often leads to model over-fitting. Any multiclassifier model can be used as a predicting model.

3. Result

For this research, we used medical recommendations from EHRs provided by the Almazov National Medical Research Centre, Saint Petersburg, Russia. The total volume of the dataset is approximately 3 million unstructured records with 440,000 unique words.

After the naive extraction, 36,856 unique abbreviated words were obtained and reduced to 26,989 after splitting by punctuation marks. To collect the statistics for entropy, the Leo Tolstoy novel "War and Peace" with 54,294 unique words was

processed. A dataset of 48,530 natural words and 23,346 abbreviations was collected to choose the threshold. The natural word classifier accuracy was 89% with 0.2 threshold.

The resulting dictionary contains 4,658 unique abbreviations after discarding words whose entropy was above 0.2. Unfortunately, some abbreviations like "АД" (short for "blood pressure", but the word "ад" is translated as "hell") have very large entropy and cannot be divided from natural words, because it is already a separate word.

The recommendation records were normalized using pymorphy2 [6] and stop words were filtered using the NLTK Python module. Unencrypted medical terms were identified in the domain dictionary in the recommendation corpus. The result was 224,307 texts with 19,980 unique terms. The validation set contained 30% of data. The selected domain dictionary was the "Encyclopedic dictionary of medical terms". TF-IDF context vectors (context_window=5, max_features=200) were counted. The projections of the context feature vectors with t-distributed Stochastic Neighbor Embedding (t-SNE) [7] are shown in Fig. 2. As can be seen, TF-IDF vectorizers grouped by context together even if the abbreviation columns in the features were skipped.

After applying difference models to the validation dataset, the Support Vector Classifier emerges as the best model. Its accuracy is 94.5%, ROC-AUC score is 99.7%, and F1 score is 93.7% (Tab. 1).

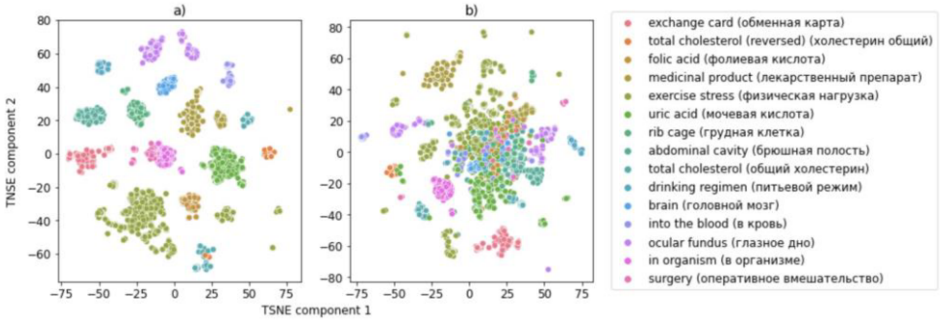

Figure 2. Context feature maps for top 15 terms a) with abbreviation columns, b) without abbreviation columns.

Table 1. Results of abbreviations decryption.

Model	Accuracy	ROC-AUC score	F1 score
RandomForestClassifier	0.938	0.968	0.933
LogisticRegression	0.939	0.985	0.930
XGBClassifier	0.939	0.973	0.935
SGDClassifier	0.940	0.983	0.930
CatBoostClassifier	0.940	0.986	0.932
SVC	0.945	0.997	0.937

Discussion & Conclusion

This work presents an algorithm for automatic extraction and decryption of abbreviations from medical records. To achieve this, we applied a naive hypothesis and then improved the algorithm via entropy of words. The abbreviation filtering algorithm is implemented. After testing the decryption method with randomly chosen records, the classifier shows high accuracy (94.5%).

However, this work does not consider the abbreviations that are written in lower case, for example "mmHg" (millimeter of mercury) and frequent abbreviations (they are virtually always written in their short form, so we were unable to find their context using a domain dictionary). We intend to develop more abbreviation rules and expand the naive dictionary.

In our future research, we intend to apply the algorithm to different corpora, especially where abbreviations are ambiguous. This module is aimed to help data scientists improve their models that use free-form records for predicting processes associated with healthcare.

Acknowledgments

This research is financially supported by The Russian Science Foundation, Agreement #19-11-00326

References

[1] Gamal A, Barakat S, Rezk A. Standardized electronic health record data modeling and persistence: A comparative review. Journal of Biomedical Informatics 2021; 114.

[2] Funkner AA, Egorov MP, Fokin SA et al. Citywide quality of health information system through text mining of electronic health records. Applied Network Science 2021; 6.

[3] Wen Z, Lu XH, Reddy S. MeDAL: Medical Abbreviation Disambiguation Dataset for Natural Language Understanding Pretraining. In Proceedings of the 3rd Clinical Natural Language Processing Workshop. Association for Computational Linguistics; 2020.

[4] Zweigenbaum P, Deleger L, Lavergne T, Neveol A, Bodnari A. A Supervised abbreviation resolution system for medical text. CEUR Workshop Proceedings 2013; 1179.

[5] Shilov IM. Automatic identification and decryption of abbreviations in the text [dissertation on the Internet]. Saint-Petersburg: Saint-Petersburg State University; 2016. Available from: https://nauchkor.ru/pubs/avtomaticheskoe-vyyavlenie-i-rasshifrovkaabbreviatur-i-sokrascheniy-v-tekste-587d36365f1be77c40d58984

[6] Korobov M. Morphological analyzer and generator for Russian and Ukrainian languages. In: Khachay MY, Konstantinova N, Panchenko A, Ignatov DI, Labunets VG (Edrs.) Analysis of images, social networks and texts. Berlin: Springer; 2015.

[7] Laurens M. Accelerating t-SNE using Tree-Based Algorithms. Journal of Machine Learning Research 2008; 15:3221-3245.