Lemmatization of Russian Language by Tree Regression Models

I.Akhmetov[1[0000-0002-3221-9352]], A.Krassovitsky [1[0000-0003-2948-374X]], I.Ualiyeva [1[0000-0003-3853-8896]], and R.Mussabayev [1[0000-0001-7283-5144]]

[1] Institute of Information and Computational Technologies,
Pushkin str. 125, Almaty, Kazakhstan
i.akhmetov@ipic.kz

Abstract. In this article, we consider the problem of supervised morphological analysis using an approach that differs from industry spread analogs. The article describes a new method of lemmatization based on the algorithms of machine learning, in particular, on the algorithms of regression analysis, trained on the open grammatical dictionary of Russian language. Comparison of obtained results was performed with existing alternative applications that are used nowadays for addressing lemmatization problems in NLP problems for Russian language.

Keywords: lemmatization, text normalization, supervised machine learning, decision tree regression models.

1 Introduction

A common problem in the analysis of texts is a large feature space that corresponds to the dictionary used in text vectorizers (90-200 thousand attribute entities). A common approach to reduce vector space is to normalize texts. It shows considerable success in reducing vector space in cases when a relatively small amount of available text of datasets leads to more balanced and inaccurate models. In addition to dimensionality reduction of Vector Models it also reduces the size of the index which speeds up all text processing operations.

Normalization, in particularly, word lemmatization is a one of the main text preprocessing steps needed in many NLP problems as well as in practical applications. The lemmatization is a process for assigning a lemma to each word. Lemma is a canonical (normal, dictionary) form of the lexeme. For instance, in Russian language, the normal form of a noun corresponds to the form in the nominative case in the singular (for example, *sestry / sisters)* → *sestra / sister*), for adjectives, the normal form will correspond to the nominative case, masculine, singular (*sil'nymi / by strong)* → *sil'nyj / strong*), verbs have the normal form corresponding to the infinitive (*begut / they run)* → *bezhat' / to run*) [1]. A number of approaches exists for lemmatization [2–4]. While it is not so bright for Russian. Our machine learning task is complicated by that fact that Russian is influenced by a number of essential attributes related to the internal complexity of this natural language [5].

Two popular morphological analyzers for the Russian language are the *pymorphy2* [6] and *MyStem* [7], the comparison with which is carried out in this article. MyStem is a tool for morphological data acquisition for Russian languages, pymorphy2 is a morphological analyzer for Russian and Ukrainian languages. They both are freely available for non-commercial and limited commercial use. MyStem is based on dictionary, automatically converted to *trie* structure. Pymorphy2 is based on OpenCorpora dictionaries [8]. Both of them are based on hand-written set of heuristic rules, and on corpus statistics to eliminate extra morphological variants and obtain morphology of a wide lexical coverage.

In our work the lemmatization is treated by building tree regression models [9] i.e., by supervised automatic learning with decision trees that are constructed corresponding to language grammatical features. A number of regression models have been compared by training on a well-built dictionary. Our method is a direct supervised approach of building word lemma regressor. In principle, this approach may be applied to any language, that captures the property of high variability inside its syntactic forms. Our approach estimates the possibility of computing syntactic models using only datasets of big data dictionaries.

The article shows a comparative analysis of the lemmatization by pymorphy2, MyStem and the new method proposed by the authors. For testing purposes lemma data set from is obtained by parser of ABBYY Compreno [10]. ABBYY tool is taken as a gold standard of comparison approach, because nowadays is considered as state of art for the industrial techs. Dataset contains 225 publications taken from the Kazakhstan news portal *tengrinews.kz* marked by this parser. The proposed lemmatization can be used in various fields; however, it is currently being considered for the preprocessing of Russian-language media texts. The motivation to develop the lemmatizer is due to the fact that the same entities of Russian language used in Kazakhstani media are partially different from the entities of Russian language used in Russian media.

2 Regression models

2.1 Dataset

For training models, the open grammatical dictionary of Russian language [11] was used, consisting of more than 2 million words and their normal forms. To test the method, the corpus of the Kazakhstan news portal *tengrinews.kz* was taken, including 225 publications (20621 words). All publications are parsed via the ABBYY parser. For obtaining accuracy of the regression models open-corpora dataset [8] is used.

2.2 Method

Vectorization of words is performed character-by-character into a vector of fixed length 30 (feature space) and values as a sequence of a letter in the alphabetical order with following zeros. After vectorization of various word forms and their initial forms obtained from the open dictionary, two arrays of vectors were obtained, which were randomly divided into training and test samples in a ratio 70 to 30. The resulting arrays were fitted into corresponding regression models. The following regression models were used: Decision Tree, Random Forest, Extra Tree, and Bagging from sklearn Python library [12]. Each feature receives weights according to its contribution to computed lemma. See Fig.1., where X and Y axes means feature and weight, correspondingly. Extra Tree achieves 88/81%, Random Forest Regressor achieves - 80/76%, Decision Tree Regressor - 73/68% and Bagging Regressor - 80/76% on train/test accuracy scoring. Experiments with variations on hyperparameters of the computation algorithms have shown that their optimization (i.e., a search for optimal values of tree depth and maximal splitting size) does not give essential improvement. The performed learning and test results are presented in Tables 1 and 2.

Table 1. Accuracy of tree regression models.

Regressor	Crossvalidation (num. of folds=5, 2337988 words)	Train/Test split (67/33, 2337988 words)		ABBYY corpus check (20621 words)	Open-corpora check (347409 words)
		Train	Test		
DecisionTree	0.3466	0.7296	0.6788	0.7561	0.7088
RandomForest	0.5991	0.8035	0.7562	0.3623	0.3556
ExtraTree	0.6697	0.8759	0.8096	0.7544	0.6840
BaggingRegressor	0.6006	0.8045	0.7571	0.3682	0.3569

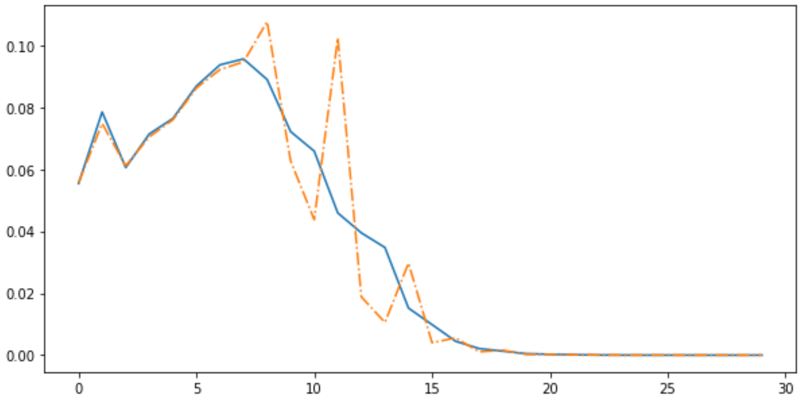

Fig. 1. The weights for features distribution (feature importance) reflect Russian word morphology (prefix, root, suffixes, etc.) are shown for Random Forest (dotted line) and Extra Tree (continues line) regression models.

Table 2. Alternative lemmatizators.

Lemmatizator	The whole data set 2337988 words	ABBYY corpus check (20621 words)	Opencorpora check (347409 words)
Pymorphy2	0.7643	0.8181	0.8967
MyStem	0.6488	0.7250	0.8208

2.3 Experiments

In order to evaluate the performance of the method, the authors' lemmatizer was compared with the MyStem and pymorphy2 lemmatizers, using the ABBYY parser to provide the testing data set. The number of wrongly lemmatized words is compared and shown by Venn diagram for these three lemmatizers by using the ABBYY test dataset (see Fig.2).

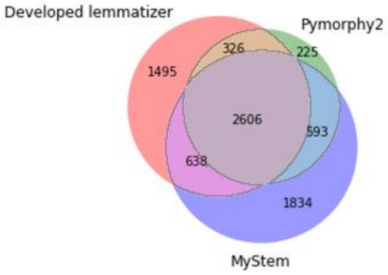

Fig. 2. The total number of errors and number of mutual errors in the testing dataset(20621 words) for our lemmatizer, MyStem and pymorphy2 are shown.

Conclusion

Decision tree regressors is not a silver bullet in machine learning, yet it can be a good tool in modelling language models in cases when it is too complicated to compose thousands of different rules. Our approaches estimate the possibility of computing syntactic models using only datasets of big data dictionaries. As an interesting by-product it is worth to mention that we get a possibility of evaluating the amount of contextual word dependencies in a language to be explored. (e.g., by estimates from [10], the ambiguity doesn’t cost more than 10 - 20% of the index size for Russian.)

Number of experiments shown the developed new lemmatizer is able to solve the problem of lemmatization (especially for specific text topics), although it needs further training. Experiments can be continued for the corpus with a large number of publications and with the study of the speed of the algorithms.

The working version of the lemmatizer can be found at http://isa1.pythonanywhere.com/

Acknowledgments

This research is conducted within the framework of the grant num. BR05236839 "Development of information technologies and systems for stimulation of personality's sustainable development as one of the bases of development of digital Kazakhstan".

References

1. Smirnov, I.V.: Vvedenie v analiz estestvennyh jazykov. Rossijskij universitet druzhby narodov, Institut sistemnogo analiza RAN, Moskva (2014).
2. Dave, R., Balani, P.: Survey paper of Different Lemmatization Approaches. In: International Journal of Research in Advent Technology Science and Technology. Special Issue 1st International Conference on Advent Trends in Engineering, Science and Technology “ICATEST 2015.” pp. 366–370 (2015).
3. Ozturkmenoglu, O., Alpkocak, A.: Comparison of different lemmatization approaches for information retrieval on Turkish text collection. In: INISTA 2012 - International Symposium on Innovations in Intelligent Systems and Applications (2012).
4. Plisson, J., Lavrac, N., Mladenić, D.D.: A rule based approach to word lemmatization. Proc. 7th Int. Multiconference Inf. Soc. (2004).
5. Plungjan, A., Ljashevskaja, O.N., Sichinava, D.V.: O morfologicheskom standarte korpusa sovremennogo russkogo jazyka. Nauchno-tehnicheskaja informacija. Serija 2. Informacionnye processy i sistemy. pp. 2–9 (2005).
6. Korobov, M.: Morphological analyzer and generator for Russian and Ukrainian languages. In: Communications in Computer and Information Science. pp. 330–342 (2015).
7. Segalovich, I.: A Fast Morphological Algorithm with Unknown Word Guessing Induced by a Dictionary for a Web Search Engine. In: MLMTA (2003).
8. «Otkrytyj korpus» (OpenCorpora), http://opencorpora.org/, last accessed 2019/01/21.
9. Geurts, P., Ernst, D., Wehenkel, L.: Extremely randomized trees. Mach. Learn. 63(04), pp. 3–42 (2006).
10. ABBYY Company, https://www.abbyy.com/ru-ru/science/technologies/compreno/, last accessed 2019/01/21.
11. Otkrytyj grammaticheskij slovar' russkogo jazyka, http://odict.ru/, last accessed 2019/01/21.
12. Bird, S., Klein, E., Loper, E.: Natural Language Processing with Python. 1nd edn. O’Reilly (2017).