2022 IEEE 10th Jubilee International Conference on Computational Cybernetics and Cyber-Medical Systems (ICCC 2022) | 978-1-6654-8177-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICCC202255925.2022.9922818

Evaluation of Deep Learning-based Authorship Attribution Methods on Hungarian Texts

Laura Gulyás Oldal
John von Neumann Faculty of Informatics
Óbuda University
Budapest, Hungary
laurago9826@stud.uni-obuda.hu

Gábor Kertész
John von Neumann Faculty of Informatics
Óbuda University
Budapest, Hungary
kertesz.gabor@nik.uni-obuda.hu

***Abstract*—The range of text analysis methods in the field of natural language processing (NLP) has become more and more extensive thanks to the increasing computational resources of the 21st century. As a result, many deep learning-based solutions have been proposed for the purpose of authorship attribution, as they offer more flexibility and automated feature extraction compared to traditional statistical methods. A number of solutions have appeared for the attribution of English texts, however, the number of methods designed for Hungarian language is extremely small. Hungarian is a morphologically rich language, sentence formation is flexible and the alphabet is different from other languages. Furthermore, a language specific POS tagger, pretrained word embeddings, dependency parser, etc. are required. As a result, methods designed for other languages cannot be directly applied on Hungarian texts. In this paper, we review deep learning-based authorship attribution methods for English texts and offer techniques for the adaptation of these solutions to Hungarian language. As a part of the paper, we collected a new dataset consisting of Hungarian literary works of 15 authors. In addition, we extensively evaluate the implemented methods on the new dataset.**

***Index Terms*—authorship attribution, authorship analysis, stylometry, deep learning**

I. Introduction

Manual authorship attribution methods can be traced back to the 15th century, where the humanist Lorenzo Valla proved that the donation letter to Donatio Constantini was a forgery [1]. Furthermore, many of Shakespeare's dramas are proven to be written with the collaboration of other authors, such as *The Noble Kinsman* and *VIII. Henry* dramas, much of which are written by John Fletcher. The collaboration of the two authors was only an assumption of linguists until proven by automated analysis of authorial style markers [2]. Thus, authorship analysis was first used to examine the authors of literary texts, however, these methods are used in the fields of criminal linguistics, continuous authentication, plagiarism detection, examination of online messages, etc.

Stylometry encompasses all methods of text categorization based on style, be it categorization by author, genre, or even era. Exactly what we mean by "style" and which parts or properties of the text make up the author's style elements are not entirely obvious, as this may vary depending on the subject matter, era, age of the author and other factors. With the evolution of natural language processing and new machine learning techniques, it is possible to extract features that go unnoticed to the human eye [3].

Traditional statistical methods provide extremely accurate results in the field of authorship analysis, but their major drawback is the strong relation of the model design and the data structure. In contrast, deep learning-based solutions automatically discover the author's fingerprint [3].

Deep learning-based solutions have been designed for authorship attribution of English texts, but are not directly applicable on Hungarian texts. The further discussed methods operate on character-level and word-level. Word-level data in the introduced solutions are word embeddings and syntactic labels obtained from a morphological parser.

Hungarian language is morphologically rich contrary to the English language, which means that the syntactic information of a unit is expressed at word-level [4]. As a consequence, the syntactic relations in a sentence cannot be expressed with the same tags in both languages. In addition, Hungarian language has an extended alphabet of 40 letters, some of which are multi-character letters. Furthermore, word embeddings trained on English corpora cannot be applied on Hungarian words. In this paper, we provide a solution for some of these issues, which will be further discussed in other chapters.

II. Related work

Style markers can be divided into four different categories: lexical, syntactic, semantic and content-dependent style markers [5]. In this paper, the lexical and syntactic features are used for authorship attribution.

The lexical features are considered to be the simplest and most intuitive style markers, such as word frequencies, vocabulary richness, character n-grams, etc. The text, which is divided into tokens cannot be interpreted by an algorithm in its original form, the tokens are represented as a series of vectors. Several methods exist for vectorization of tokens, some of which are one-hot encoding, distributional semantic models, embeddings, etc. In the case of one-hot encoding, the tokens are represented as categorical data. The main drawback of this method is the problem of dimensionality, as it produces sparse vectors with large dimensions [6]. The distributional models create a high-dimension vector space through a statistical analysis of context in which the tokens occur [7]. The problem

of the dimensionality is not solved with distributional models, however, the cosine distance of words used in a similar context is reduced, vectors of similar words are closer, thus, words are not treated as categorical data in contrary to one-hot encoding. The most common method used for word represantions are word embedding models. Word embedding models are deep learning-based models, that eliminate the problem of large dimensionality and produce a dense vector representation. The vector coordinates of word embeddings are inherited from the weights of the hidden layer of the word embedding model [8]. Transfer learning is often used in word vectorization tasks, as the generic meaning and context of words is usually uniform across different topics. The pretrained word embedding models are further trained with the training data of a particular task for adapting the vector space [9], [10].

The basic idea of using syntactic features is that people usually subconsciously form sentences with the same syntactic structure. Compared to lexical features, the extraction of syntactic style markers require a deeper, language-dependent analysis, which is achieved with morphological analysers and text parsers. Some commonly used syntactic features are POS (*Part-of-speech*) tags, constituency trees and dependency trees. Part of speech (POS) reveals the role of the word within a grammatical structure, for e.g. noun, verb, etc. POS tags mark up a word in a corpus corresponding to a POS, reflecting its role in a sentence. In addition to POS tags, parse trees are often used as syntactic features, which are structures that highlight the syntactical relations of words or sub-phrases according to a formal grammar. Main types of parse trees are constituency and dependency trees. While constituency trees extract the hierarchical structure of a sentence, dependency-trees capture the dependencies between words in a sentence. Dependency tree is often represented as a directed graph, where the nodes represent the words and the direction of the connection expresses whether a certain node is a child or parent of the connected node [11].

POS taggers, dependency and constituency parsers cannot be used for cross-language parsing, as these tools use the grammar of a certain formal language. This is the main drawback of using syntactic style markers for authorship attribution of Hungarian texts, as designing such a method requires a morphological analyzer developed for Hungarian language.

The methods used in natural language processing have undergone rapid development with the advent of machine learning [12]. Much of the initial solutions are based on statistical methods that use lexical features. Initially, solutions were developed that performed the analysis by examining a single lexical characteristic. Later, multidimensional statistical methods appeared, which performed more complex and thus more accurate analysis. Such procedures include cluster analysis, Support Vector Machines, Nearest Shrunken Centroids, Burrows Delta [3]. The procedures have been very successful in the field of authorship attribution, however, the precise design of the features has posed major obstacles to achieving further progress. In case of these methods, the structure of the data highly impacts the design of the model. These difficulties are overcome by deep neural networks, where there is no need to design features, the network itself learns the appropriate representation from the data using the many hidden layers [13], [14]. For these reasons, we analyse and review methods implemented with deep neural networks.

A. Character-level models

Convolutional neural networks have been very successful in processing imagery data, but they are also excellent for one-dimensional structures. Patterns can be discovered on such data, thus convolutional neural networks are often used for authorship attribution [15]. Convolutional neural networks for text classification were first used by Kim [16]. The method used static and non-static word embeddings of texts as input data. Zhang et al [15] were the first to use character-level convolutional neural networks for text classification. [15]. Zhang et al. proposed two different models, one with a larger and one with a smaller convolutional filter. They further evaluated the method examining the following criteria: should we treat uppercase and lowercase characters as different data or not. They compared the proposed method with other text classification methods on the same datasets, including existing non-character convolutional models and the most common statistical methods used in natural language processing, such as the bag-of-words and bag-of-ngrams model. In the proposed method Zhang et al. used section length of 1014 characters and an alphabet consisting of 70 characters, where the characters were vectorized as one-hot encodings. Zhang et al. used multiple convolutional layers with the same number of filters, but different kernel sizes. They used 256 filters in the smaller model, while the larger model contained 1024 convolutional filters. Following the convolutional layers, they used a max-pooling method, the result of which is the input of fully-connected layers. They also used 2 dropout modules for the purpose of regularization [15]. The methods were evaluated on 6 different datasets. Most datasets contained newspaper articles or forum questions, and the use case was topic classification. For the measure of method effectiveness, the errors were used. The method performed best on larger datasets. Traditional statistical methods outperform Zhang et al. method on smaller datasets, however, on larger datasets the method proved to be significantly more accurate.

Ruder et al. [17] evaluated and compared the method of Kim [16] and Zhang et al. [15] for the task of authorship attribution. The datasets used for evaluation consisted of emails, movie reviews, blogs, twitter posts of 10 and 50 authors. The maximum length of the data was set to 500 words or 3000 characters. For the measure of effectiveness, the F1 score was used. From all of the convolutional models reviewed, the character-level CNN produced the best results and outperformed word-level CNNs with an average F1 score of 65.02% for 50 authors.

B. Syntactic models

POS tags are an excellent style marker for authorship attribution, as they capture the sentence formation patterns

of an author. Some methods using this type of data are the methods of Hitschler et al. [18] and Jafariakinabad et al [19]. Hitschler et al. proposed a method where the model is word-level and its inputs are concatenated vectors of POS tags and static word embeddings. Their basic idea was to use the POS tags and the most frequent n words of the language in conjuction as an input of a convolutional network to capture which words are often used by the author in what form [18]. The model was based on the design of Kim's model [16] with a few modifications the most significant of which are the elimination of an extra channel and the use of not only the word representation, but POS tag as well. The method uses the *Stanford* POS tagger as morphological analyzer and the POS tags are represented as one-hot encodings. The word embeddings of most frequent n words are used, while other words are treated as unknown. The examined n thresholds are 1,000, 5,000, 10,000, 50,000, N/A. The first layer of the network is an embedding layer, which is used to reduce the dimensionality of the data and capture word similarities. The next layer is a convolutional layer consisting of 100 filters with 3 different kernel sizes. Dropout module was used on the layer for regularization. The convolutional layer is followed by a max-pooling module, subsequently, fully-connected layers. One of the datasets used for evaluation is the *PAN 2012* dataset, containing writings of 14 authors. The data is divided into segments of 1,500 words, with shorter segments discarded. The proposed method achieved the best results with a threshold of n = 50,000 word frequencies. Surprisingly, using the word embeddings of the full dictionary results in less accurate results. However, generally increasing the value of n results in greater accuracy among the examined n values. The accuracy of the method on the *PAN 2012* dataset is 78.57%.

Jafariakinabad et al. [19] proposed a model whose inputs are only the extracted POS tags. In contrary to the previous method, Jafariakinabad et al. treats POS tags as a time-series data. The model allows a hierarchical analysis of the document, as the inputs of the method are POS tags grouped into sentences. The method uses the POS tagger of the NLTK package. The sentence representation is extracted with a CNN or LSTM encoder, where the former learns short-term and the latter long-term dependencies. The sentence encoder is a bidirectional LSTM unit, which extracts the author-specific syntactic patterns of a sentence. Finally, with an attention mechanism, the sentences, which most represent the style of an author are rewarded giving document representation as a result. [19]. The dataset used for evaluation is the 14-author dataset of the *PAN 2012* project, where the data is divided into 100 sentence long segments. The model achieved an accuracy of 78.76%.

C. Word-level models

Chen et al. [20] propose methods that similarly to the previously discussed method, treats the text as time series data, thus LSTM and GRU networks are designed. One of the suggested methods uses the static word embedding of words as input data, while the other uses a sentence representation. In the second method, the sentence representation is a vector containing the average corresponding values of word embeddings within a sentence [20]. The method was evaluated on the *C50* dataset, which contains corporate or industrial newswire stories of 50 different authors. The accuracy of the method on the mentioned dataset was 69.1% [20].

III. Methodology

In the previous chapter we presented the working principle of deep learning-based methods. As our research goal is the authorship attribution on Hungarian texts, we evaluated some of the discussed methods on a Hungarian dataset. The process is discussed in this chapter.

A. Dataset

There are no available annotated Hungarian corpora suitable for authorship attribution. For this reason, we created our own dataset containing literary works of 15 Hungarian authors. The data collection, cleaning and labeling is automated, we developed a tool specifically for this task. The source of our dataset is the *Magyar Elektronikus Könyvtár* (MEK), which is the oldest Hungarian electronic library. The designed tool, given a set of authors searches literary works in HTML format and downloads them if available. As the actual text of a writing is usually displayed in a distinct style, noise filtering is easily achieved. For example, title, author, publisher information are displayed with bold style, contents table items are *a* tags and they have an *href* attribute, page numbers are aligned to center of the page, while the actual prosaic text is justified. Using these observations, we create a set of rules, which exclude texts of a certain style from the HTML page of the literary work.

Only original, single-author literary works are downloaded excluding poems and dramas. The full dataset consists of 155 MB size textual data of 183 authors. However, after data balancing, the available text decreased dramatically. Because of this, we use only the 15 most prolific authors for evaluation. The size of the reduced dataset is 58 MB, and it consists of roughly 200 100-sentence length sections per author.

B. Method evaluation process

The methods are implemented in python in *keras*, *tensorflow* frameworks. We use an early stopping mechanism to avoid overfitting. As a consequence, the models are taught through different number of iterations. Hereinafter, as accuracy, we refer to the validation accuracy achieved in the iteration where the validation loss is the lowest. The training and validation data rate is 9:1. The methods were evaluated considering section lengths of 5, 30 and 100 sentences.

C. Character-level methods

We implemented the method of Zhang et al. [15] and adapted for Hungarian language. As the method is based on a character-level convolutional architecture, only minor changes were made on the original implementation. The alphabet of the English language is different from the Hungarian alphabet

and it contains multi-character letters. Therefore, the alphabet was extended and we replaced the multi-character letters with special characters, thus all Hungarian letters are encoded by different characters. Another adaptation is an optimization of the original method, which is representing the characters as embeddings as opposed to one-hot encodings. This way, vowel and consonants are usually far apart in the vector space, which captures the context of the characters.

D. Syntactic methods

We implemented the method of Jafariakinabad et al, which uses the POS tags of words grouped into sentences as input data. As the POS taggers are designed for a certain language, the POS tagger used by Jafariakinabad et al. cannot be used on Hungarian texts. As a consequence, the feasibility of syntactic methods depend on availability and quality of Hungarian morphological analyzers. In addition, as the formal grammars of Hungarian and English language are very different, separate tag sets must be used for POS tagging. We used the *magyarlanc* [21] toolkit for POS tagging and dependency parsing. The tool outputs the following data for each word respectively: Index in a sentence, original form of the word, lemma, POS tag, additional morphological properties, id of the parent node, dependency label [21]. The tagset used by the toolkit is the UD (*Universal Dependencies*) tagset, while Jafariakinabad et al. use the tagset of *Penn Treebank*. With UD annotations, a POS tag and additional morphological properties are assigned to each word. In the case of *Penn Treebank* tags, each word is charactirezed by only POS tags. The difference of the two approaches can be attributed to different degrees of morphological richness between Hungarian and English language. As the mentioned tagsets have different sizes and approach to tagging words, in our method we use the POS tag in conjuction with the dependency label extracted with the *magyarlanc* toolkit. The model of Jafariakinabad et al. does not require further adjustments for applying the method on Hungarian texts.

IV. Results

The evaluation process of Zhang et al's method on Hungarian texts is displayed on Fig. 1, where N is the length of input data. The method showed greatest accuracy on shorter text segments of 5 sentences. Increasing the size of the input results in less accurate results. The achieved accuracies for 5, 30 and 100 sentence length sections are respectively 52.42%, 35.37%, 34.28%.

The process of training the method of Jafariakinabad et al. on Hungarian texts is shown on Fig. 2. The method proved to be more accurate on longer segments of data, increasing the size of the input results in greater accuracy. The achieved accuracies for 5, 30 and 100 sentence length sections are respectively 35.95%, 59.74%, 59.31%.

The process of training is displayed on Fig. 2. The method achieved the greatest accuracy of 59.74%, 59.31% on longer text segments of 30 and 100 sentence lengths respectively. The results of the evaluation are displayed in Table I, where

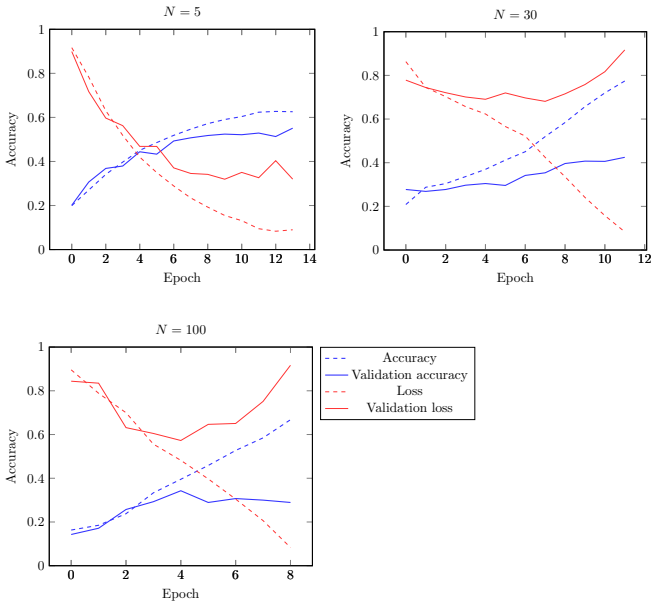

Fig. 1: Training, validation accuracy and loss during training of Zhang et al. method

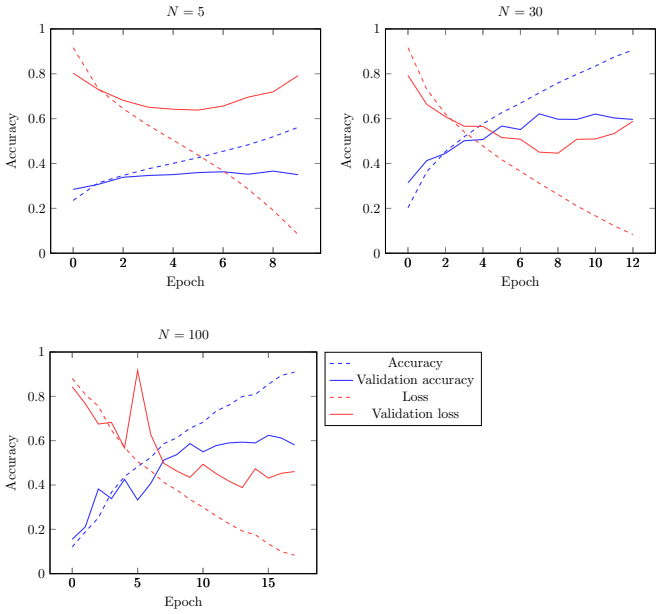

Fig. 2: Training, validation accuracy and loss during training of Jafariakinabad et al. method

Z refers to the method of Zhang et al, while J refers to the method of Jafariakinabad et al.

V. Discussion

The discussed methods are reviewed with respect to:

- ***Style markers and required tools***
  The main advantage of character-level solutions is the language-independent design of the model. As long as a character represents a letter or other symbols, the method

TABLE I: Results of method evaluation on Hungarian data

	N = 5		N = 30		N = 100	
data size per author	3900		653		199	
method	**Z**	**J**	**Z**	**J**	**Z**	**J**
number of epochs	14	10	12	13	9	18
accuracy (%)	**52.42**	35.95	35.37	**59.74**	34.28	**59.31**

is easily adapted to other languages. This is not the case with syntactic models, as they require a language specific morphological analyzer.

- ***Neural network architecture and training circumstances***
  Zhang et al. use convolutional neural networks, while Jafariakinabad et al. use GRU and LSTM units in their model. Convolution consists of simple algebraic operations, thus parallel execution of operations is extremely effective. This is not the case with LSTM or GRU units, as temporal dependencies of the input data requires sequential execution. This causes the training to be much slower on a graphics card as opposed to convolutional neural networks [22].
- ***Length of input data***
  The method of Zhang et al. outperforms Jafariakinabad et al. syntactic model on shorter segments of 5 sentences, while the syntactic model achieved greater accuracy on longer data of 30 and 100 sentences.
- ***Dataset size***
  The effectiveness of some methods lies in the use of extremely large datasets, which is the case with the method of Zhang et al. As there are no available large Hungarian datasets for authorship attribution, it is unknown how increasing the dataset size affects the effectiveness of methods. Jafarikinabad et al. evaluated their method on a much smaller dataset compared to Zhang et al. and achieved similar results.
- ***Accuracy***
  The method of Zhang et al. achieved the best accuracy of 52.42% on smaller text segments, which were 5 sentence long as displayed in Table I. On longer text segments, the method of Jafariakinabad et al. outperformed the character-level method. The achieved accuracy on 30 sentence long segments were 59.74%-t, while using 100 sentence length segments it was 59.31%.

VI. Conclusion and future work

In this paper we introduced the state of the art methods for authorship attribution and propose adjustments for adaptation to morphologically rich languages, such as Hungarian. We reviewed methods that operate on characters, syntactic units and words. In the paper we evaluate some of the analysed methods on an automatically obtained annotated Hungarian dataset of literary works collected with a tool designed specifically for this task.

The evaluated methods of Zhang et al. and Jafarikinabad et al. show best results considering different aspects. The method of Zhang et al. achieved best accuracy on small data segments, while the method of Jafariakinabad et al. showed best results on longer texts.

Our future work is focused around the evaluation and adaptation of methods, which use word embeddings as input data, such as the method of Hitschler et al. and Chen et al. Our long-term future work will be centered around open-set authorship verification with metric-learning methods.

Acknowledgments

The research was carried out with the support of the Ministry of Innovation and Technology from the National Research, Development and Innovation Fund, within the framework of the New National Excellence Program ”ÚNKP-21-2”. The authors acknowledge the support of the National Talent Program under the NTP-SZKOLL-21-0038 and NTP-HHTDK-22-0022 projects.

The authors are grateful to the members of the Applied Machine Learning Research Group of Obuda University John von Neumann Faculty of Informatics for constructive comments and suggestions.

References

[1] S. Dobi Jan, T. Mészáros, and M. Kiss, “Shtylo: stilometriai elemzések webes támogatása,” in *Magyar Számítógépes Nyelvészeti Konferencia*, 2018, pp. 423–436.

[2] P. Plecháč, “Relative contributions of shakespeare and fletcher in henry viii: An analysis based on most frequent words and most frequent rhythmic patterns,” *Digital Scholarship in the Humanities*, vol. 36, no. 2, pp. 430–438, 2021.

[3] M. Kiss, “Stilometriai elemzés lehetőségei magyar történeti szövegkorpuszon,” *DIGITÁLIS BÖLCSÉSZET*, vol. 2, pp. 15–33, 2019.

[4] R. Tsarfaty, D. Seddah, Y. Goldberg, S. Kübler, Y. Versley, M. Candito, J. Foster, I. Rehbein, and L. Tounsi, “Statistical parsing of morphologically rich languages (spmrl) what, how and whither,” in *Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically-Rich Languages*, 2010, pp. 1–12.

[5] E. Stamatatos, “A survey of modern authorship attribution methods,” *Journal of the American Society for information Science and Technology*, vol. 60, no. 3, pp. 538–556, 2009.

[6] M. A. El Affendi and K. Al Rajhi, “Text encoding for deep learning neural networks: A reversible base 64 (tetrasexagesimal) integer transformation (rit64) alternative to one hot encoding with applications to arabic morphology,” in *2018 Sixth International Conference on Digital Information, Networking, and Wireless Communications (DINWC)*. IEEE, 2018, pp. 70–74.

[7] S. Evert, “Distributional semantic models,” in *NAACL HLT 2010 Tutorial Abstracts*, 2010, pp. 15–18.

[8] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” *arXiv preprint arXiv:1301.3781*, 2013.

[9] K. Weiss, T. M. Khoshgoftaar, and D. Wang, “A survey of transfer learning,” *Journal of Big data*, vol. 3, no. 1, pp. 1–40, 2016.

[10] S. R. Chandaran, G. Muthusamy, L. R. Sevalaiappan, and N. Senthilkumaran, “Deep learning-based transfer learning model in diagnosis of diseases with brain magnetic resonance imaging,” *Acta Polytechnica Hungarica*, vol. 19, no. 5, 2022.

[11] M. Zhang, “A survey of syntactic-semantic parsing based on constituent and dependency structures,” *Science China Technological Sciences*, vol. 63, no. 10, pp. 1898–1920, 2020.

[12] A. Pejić and P. S. Molcer, “Predictive machine learning approach for complex problem solving process data mining,” *Acta Polytechnica Hungarica*, vol. 18, no. 1, pp. 45–63, 2021.

[13] L. Deng and Y. Liu, “A joint introduction to natural language processing and to deep learning,” in *Deep learning in natural language processing*. Springer, 2018, pp. 1–22.

[14] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
[15] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," *Advances in neural information processing systems*, vol. 28, pp. 649–657, 2015.
[16] Y. Kim, "Convolutional neural networks for sentence classification," 2014.
[17] S. Ruder, P. Ghaffari, and J. G. Breslin, "Character-level and multi-channel convolutional neural networks for large-scale authorship attribution," *arXiv preprint arXiv:1609.06686*, 2016.
[18] J. Hitschler, E. Van Den Berg, and I. Rehbein, "Authorship attribution with convolutional neural networks and pos-eliding," in *Proceedings of the Workshop on Stylistic Variation*, 2017, pp. 53–58.
[19] F. Jafariakinabad, S. Tarnpradab, and K. A. Hua, "Syntactic recurrent neural network for authorship attribution," *arXiv preprint arXiv:1902.09723*, 2019.
[20] C. Qian, T. He, and R. Zhang, "Deep learning based authorship identification," in *Class Report for CS224n: Natural Language Processing with Deep Learning*. Stanford University, 2017.
[21] Z. János, V. Veronika, and F. Richárd, "magyarlanc: A toolkit for morphological and dependency parsing of hungarian," *Proceedings of RANLP 2013*, pp. 763–771, 2013.
[22] K. Alvarez, J. C. Urenda, O. Csiszár, G. Csiszár, J. Dombi, G. Eigner, and V. Kreinovich, "Towards fast and understandable computa-tions: Which "and"-and "or"-operations can be represented by the fastest (ie, 1-layer) neu-ral networks? which activations functions al-low such representations?" *Acta Polytechnica Hungarica*, vol. 18, no. 2, 2021.