JOURNAL OF APPLIED INTELLIGENT SYSTEMS & INFORMATION SCIENCES
VOL 1. ISSUE 2, PP. 118-127, DECEMBER 2020.
Available at: www.journal.research.fanap.com
DOI: 10.22034/jaisis.2021.269735.1018

AUTHORSHIP ATTRIBUTION IN HISTORICAL AND LITERARY TEXTS BY A DEEP LEARNING CLASSIFIER

Ehsan Reisi[1,*], Hassan Mahboob Farimani[2]
[1]*Faculty of Persian language and literature, University of Isfahan, Azadi square, Isfahan, Iran*
[2] *Center of Fundamental and Applied Studies of Mystical Literature, University of Isfahan, Azadi square, Isfahan, Iran*

ABSTRACT

One of the important problems that language and literature scholars face is the difficulty of determining the author of the historical and literary texts. Deep learning, the latest available approaches for solving such problems, provides high accuracy results. In this paper, we show how to overcome ownership claims in historical texts by deep learning methods that are designed for text classification. In this regard, we propose a convolution neural network with a four-part architecture and self-attention mechanism to classify texts. In addition, the proposed method increases the accuracy of Author determination up to 2% in comparison with existing methods. Moreover, in our case study, Khān al-Ikhwān, written by Nāsir-i Khusraw, the author determination accuracy was 86%. Although our focus is on Persian historical textbooks through this article, our method can be applied to other languages effectively.

KEYWORDS: Text Mining, Deep Learning, Authorship Attribution, Text Classification, Convolutional Neural Networks.

1. INTRODUCTION

Authorship attribution is a practical task that has many usages in forensics, fraud detection, document identification, finding the true author, and so forth. Whereas, authorship attribution can apply in many other fields like literature and history but lack of datasets and reliable data are the main problems that create obstacles to train model correctly. In current work, we shed light on one of the mysterious questions, finding the true author of Khān al-Ikhwān, by deep learning method. Our solution does not depend on any specific language and works with short texts and datasets well. Therefore, by a novel deep learning technique, we make way for researchers who work on texts and those whose data are not sufficient to use general solutions of authorship attribution.

Identifying an author of a piece is one of the most important topics in linguistic and literary research. In recent years, its application extends to other scientific branches such as criminology. To respond to these demands, new computer tools such as text mining and artificial intelligence techniques are utilized in this area.

Finding the true author of a text, also known as authorship attribution, is sometimes very complex especially for historical and literary texts. Since manuscripts or other texts have been subjected to gradual changes over time due to geographical, environmental, political, and ideological factors. For this reason, significant parts of the historical and literary texts have undergone a transformation, and we did not inherit them as originally written. For example, parts of a book have been lost or incorrectly included in another book and non-authentic parts have been added to the following texts.

In addition, referencing authors' names while copying their work was not consistently done in ancient times. Thus, the names of authors of some works are not clear, and some are mistakenly attributed to other writers.

* *Corresponding Author, Email: e.reisi@ltr.ui.ac.ir*

RECEIVED: 21 OCTOBER, 2020; ACCEPTED: 25 DECEMBER, 2020; PUBLISHED ONLINE: 28 DECEMBER, 2020

Several well-known examples of such problems exist in historical texts. For Instance, there are various narratives from Ferdowsi's Shāhnāmih, which is the most important Persian masterpiece and world's longest epic poem. Shāhnāmih versions range from 48 thousand to 67 thousand stanzas (Minavi, 2001), but it is still not clear which verses have been written by Ferdowsi himself. Another famous example is the copies of the Hadiqat al-Haqiqah (The Walled Garden of Truth) by Sanai Ghaznavi which is the first Persian mystical collection of poetry are between 3800 and 14000 stanzas (Hosseini, 2001), that is 10200 stanzas of difference. Despite the great importance and high standing of this work and its author in Persian language and literature, it is still not clear how many of these verses are certainly Sanai's. These problems are very common in other regions, the authenticity of the Corpus Caesarianum has been obscured for centuries (Kestemont, et al., 2016). Therefore, finding true author of such books is critical for researchers particularly for literary stylists.

Another challenge of our work is the scarceness of information about old texts and manuscripts. For many languages, finding word-nets is very complicated and the syntax of the language and patterns of the style of the writing have evolved over centuries. Therefore, available etymological, syntactic, and stylistic resources might not provide proper answers to the ancient texts' problems. As a result, the creation of new methods of text mining is required to address these problems.

Our case study, Khān al-Ikhwān, is a noteworthy book that we claim is written by one of the great Persian writers and poets Nāsir-i Khusraw Qubādiyāni. Nāsir-i Khusraw's poetic works and editorships are contributive in the publication and promotion of Isma'ili beliefs while occupying a very important place in Persian literature and Islamic thought. Over the centuries, many books have been referenced to him, but it is uncertain which of these works are in fact his own work. One of the most important of these works is Khān al-Ikhwān, which has already been discussed by other researchers.

There is no reference in ancient and historical texts about our confirmation of Nāsir-i Khusraw being Khān al-Ikhwān's author. However, in contemporary researches, Ivanow considered it as part of Nāsir-i Khusraw's writings (Shahidi, 1994), but Seyyed Ja'far Shahidi rejected the attribution of this work to Nāsir-i Khusraw. In this regard, Shahidi says: "For reasons that are not mentioned in this discussion, I am skeptical about Khān al-Ikhwān as Nāsir-i Khusraw's works, and maybe this book was written during or after Hasan Sabbah" (Shahidi, 1994). Shahidi has not provided any reason to prove his claim. However, there has been some doubt about the validation of the attribution of this work to Nāsir-i Khusraw so far, and none of the assumptions has been proven. Therefore, some vogue theories about the authorship of Khān al-Ikhwān are proposed. We will prove the true author of Khān al-Ikhwān is Nāsir-i Khusraw with our deep learning method.

In this study, we propose deep learning to recognize and validate Nāsir-i Khusraw's books from other texts. Our network can distinguish whether a text written by Nāsir-i Khusraw's or not. Our structure has an embedding layer that first establishes a word vector, then three layers of convolution with max-pooling with a self-attention layer. The last layer is a normal layer for classification. This neural network learns with back-propagation. We compare our method with LSTM, LSTM with CNN net, and CNN that confirms the superiority of our method over other methods. Our architecture is designed to get all important clues in the text that shows author style. These clues, which also are known as features, have two general types: local and long-range. Convolution layers can extract features and local dependency very effectively but for covering long-range we need an attention layer. Therefore, both long and short dependencies over different parts of data (sentences) are used for determining the author of the input text.

We evaluate network by gathered text with the same style or topic from four to seven-century. In this regard, we have created a dataset from manuscripts and converted them to text files. Finally, our network classifies text to determine Nāsir-i Khusraw's ownership (see Fig. 1).

The remainder of this paper is organized as follows: In the next section (see Section 2) we briefly review attribute authorization and other related methods so we try to consider the variety of methods for sake of completeness. In section 3, we demonstrate the architecture of our network and its details. Section 4 presents our experimental results, comparison with other solutions, and implementation details. Finally, in the last section, we conclude.

Fig. 1 extracting text from manuscripts and data processing procedure

2. Related Works

In this section, we present previous methods that work on authorship attribution or classify texts by considering their author and provide a review of previous works and compare them with each other. Two main categories exist for authorship attribution. The first one is based on feature engineering and classical machine learning solutions and the second is based on deep learning view. We choose the second approach due to our limits in this project.

Stylistics differentiate between various styles based on word frequency and distribution which is very general for author identification. To solve this shortcoming, Diurdeva et al. (2016), discussed a profile-based method that uses character N-gram distribution as writing style. Their approach works with distribution and does not include any semantic or word relationship of content of the books. Velásquez & Oberreuter (2013) presented word-frequency-based on stop words algorithm after segmentation of document to detect plagiarism. Such methods identify stop words or function words as a feature to detect the style of the writer (Kestemont, et al., 2016). None of these methods considers context since the style is hard to copy and deceive readers but copying the context is easy. Savoy (2012) proposed a standardized "Z score" to classify a document for specific vocabulary. Seroussi et al. (2014) considerd on-line texts like blogs for authorship attributes by using topic modeling. Several papers have used extra information around the documents to find author attribution. Li et al. (2015) provided many features for this task. These methods employ function words, stop words, or other features. Some researchers use the structure of documents for classification (Shalymov, et al., 2016). On the one hand, extracting function words need datasets that are rare for old text. On the other hand, information based on word frequency does not correlate to the contextual and semantic of a text especially a literary text. Therefore, we should consider other solutions that have the following characteristics:

i. Do not need dataset or any previous knowledge about text.
ii. Consider both context and style.
iii. Because of structure of text changes over the time, the solution must learn structure (not use constant structure).
iv. Semantic must be learnt and used as an important part of the text.
v. Not require hard pre-processing.
vi. Not require feature engineering.

The deep learning methods have more or less these properties. Considering these characteristics would assist us to create an independent solution that does not relate to specific language and context.

Since word2vec was introduced by Mikolov et al. (2013), the problem about prior knowledge has been solved. The embedding layer learns word relation efficiently and deals with new and old text adaptability. Therefore, an effective solution has been found that considers the structure, semantics without any prior knowledge and it is an unsupervised method. GloVe (Pennington, et al., 2014) is a more efficient method for word representation but it does not manage online learning. Such methods are the first step for the classification of documents and texts; however, the main problem has not been solved yet. Kalchbrenner et al. (2014)

introduced a method for representing sentences accurately by dynamic convolution neural networks that are able to work with variable sentence length. Le & Mikolov (2014) represented an unsupervised method for fixed-length texts (sentence, paragraph, document). They claimed that such a method can overcome issues with the previous methods such as bag-of-words methods because bag-of-word-based methods (and most of the solutions based only on word frequency) do not consider semantic and word ordering (as we mentioned previously). Yet, the train of deep learning does not stop. Chen (2017) shows word2vec can be used for creating sentence vectors by simple averaging on word of word embedding. The purposed method is known as Doc2VecC. Alharthi, et al. (2018) demonstrated a recommended system based on text author style. Their work is the same as Shrestha, et al. (2017). They use CNN with one-layer convolution for text classification. Ma & Hovy (2016) introduced new architecture that included LSTM, CNN, and CRF for POS tagging and NER. We compare our neural network with an architecture similar to their work. Sboev et al. (2016) used similar architecture without CRF. Hitschler et al. (2017) proposed a CNN classifier with one layer convolution, which removed words below the determined threshold and worked with their POS-tags.

We use three convolution layers in the second part of our network, this architecture was adopted from (Badrinarayanan, et al., 2017) decoder. However, our convolution layer has several differences with its encoder. For example, they use up-sampling before each convolution layer, but we employ max pool after the layers are finding the best features depend on the problem. Another similar architecture is done by Kim (2014) that only has one convolution layer with a different size filter. In comparison, our filters have the same size at each layer yet we increase the filter number after each max-pooling. Other convolution types like separable convolutions (MobileNet (Howard, et al., 2017)) depth-wise separable convolution (Xception (Chollet, 2017)), lightweight convolution (Wu, et al., 2019), and so on have a remarkable result in other task but are not compatible with our model, so, we consider them for feature works.

Although Attention Mechanism first introduced in machine vision (Mnih, et al., 2014), it is one of the innovations that has great influence in NLP especially machine translation (Bahdanau, et al., 2014). Capturing long-distance dependency, we apply our model with self-attention. Moreover, another noteworthy text (Vaswani, et al., 2017) tries to use attention as an effective tool. To weigh influential author style features that are repeated trough out a text, we apply global attention (Luong, et al., 2015). Other useful architectures are used for different proposes like Yin et al. (2016) that are very inspiring.

Furthermore, some other methods exist that model text as complex networks or use graph base models to classify texts. Amancio considers the topological properties of manuscripts to determine fake articles (Amancio, 2015a). Besides that, he develops a similar method for short texts (Amancio, 2015b). However, we do not find these methods suitable for our application. In the next section, we will describe our method.

3. Architecture

Our network contains four parts: the first part is the embedding layer that word2vec represents words as a vector, the input of this layer is a 20×100 matrix. If a sentence is longer than twenty, it will truncate, and if less than twenty, it uses zero paddings. Each sentence matrix has 100 rows that are provided by the embedding layer. By experience, increasing the size of the matrix would not necessarily improve efficiency. The second part is the three layers convolution, each layer has its own filters and properties (see Fig. 2). Enhancing long-term dependency, self-attention comes after convolutions and max-pooling. Finally, a flatten layer with two classes determines that whether the input text belong to the author or not.

Before we discuss the network structure, segmenting text to the words as an input will be explained. We segmenting part of (10%) proposed documents by Nāsir-i Khusraw, sentence by sentence. Each sentence is extracted by an expert with regards to sentence structures in Persian literature depended on the verbs in the sentences. However, sometimes verbs have been omitted due to ellipsis and quasi-sentences, interjection, and nouns. One of the main challenges in this research was to recognize the sentences of historical and literary texts. The language of these texts, in particular, the works of Nasir-i Khusraw, is from the fourth century AD, more than a thousand years ago, and therefore its grammatical and lexical structure is significantly different from those of modern language. On this basis, it is essential to make the initial recognition of the sentences of these texts by experts.

After 10% of sentences (selected randomly) are segmented by the experts, we compute the average and variance of sentences' length. As a result, we find optimal fixed-length is around twenty words for such texts. Finally, we segment text by punctuations, done by a proofreader, with a max length of twenty words. Now the text is segmented and ready to be fed into the network.

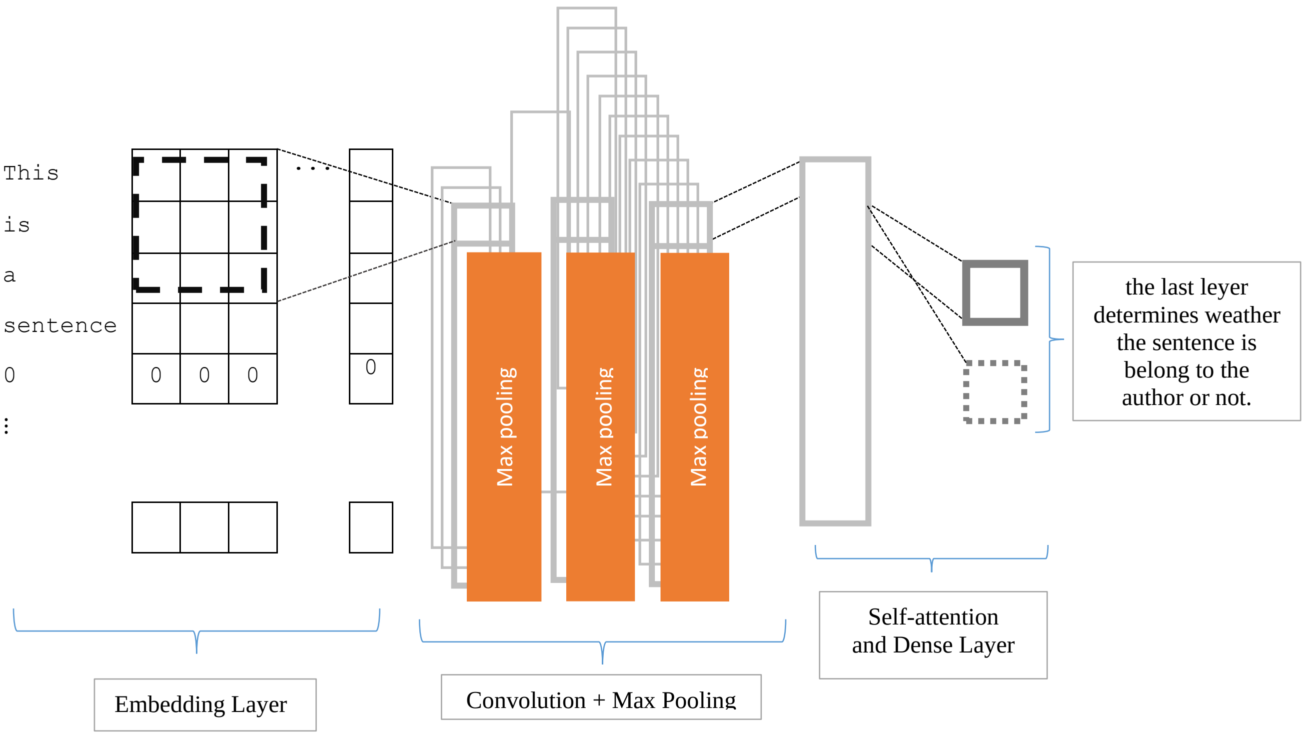

Fig. 2 Three layers architecture with max pooling and self-attention

The first part (embedding layer) manages to convert words to vector, create the input matrix $\mathbf{X} \in \mathbb{R}^{n\times m}$ with zero paddings or truncate long sentences if necessary. Each word in the corpus is converted to vector $\mathbf{x_i}$ $(0 < i \leq \mathcal{N} - 1)$, $\mathcal{N}$ Number of words in corpus with chronological order in the original sentence, *n* average length of a sentence in the corpus (explained above), and *m* depends on the embedding layer, in word2vec normally vector length is considered between 100 and 300. This layer is trained on the whole corpus separately, we call the pre-train phase, word2vec algorithm helps us to complete this conversion. After conversion, the input matrix $\mathbf{X}$ is ready for the next layer (convolution and max-pooling layer).

$$\mathbf{X} = x_{(i.n)} \oplus x_{(i.n)+1} \oplus \ldots \oplus x_{(i.n)+n-1}, \qquad 0 < i \leq \mathcal{N} - 1 \tag{1}$$

In the second part of our network, each convolution sublayer includes different size of filters $w \in \mathbb{R}^{h\times h}$ which h is the windows' size for producing new features. As an illustration, the extracted features are generated by:

$$c_i = \mathbf{W}.\mathbf{X}_{i.n:(i.n)+n-1} + b \tag{2}$$

where b and **W** are bias and weight and learnable trough out training phase. After max-pooling, features face with another convolution sublayer.

The noteworthy point is when a sentence's matrix is truncated by convolution operation to find proper features, the near relations are more important than the long relations between the words. For instance, subject and verb must match based on grammar in a sentence and it does not depend on who makes the sentence. Whereas, an adjective phrase is created by each human, to describe a sense of emotion, more distinctively because people have their own point of view. Therefore, we capture near word relations with the same small filter size. However, after each convolution layer max-pooling operation remove unnecessary features and this work cover long distance partially. By contrast, an assortment of filter size at each convolution layer makes a conflict in learning. Training the network is done by Adam optimization with backpropagation strategy.

Dropout training is an effective way to tackle overfitting and preventing feature co-adaptation. On the other hand, many dropout strategies are baffling as they have detrimental effects on performance even though using strategies like fast drop-out training introduced by Wang & Manning (2013). Therefore, we apply dropout with an alternative strategy in our network. First of all, at the input layer we use dropout with a higher rate method (Srivastava, et al., 2014) also we decrease it in the inner layers second, in the third convolution layer we use batch normalization instead of dropout (Ioffe & Szegedy, 2015) that improves our performance.

Convolution layers size and number are also critical so we apply different layers to figure out the best configuration. Generally, convolution layers number, more than three layers, are not effective. As mentioned before the size of filters magnify due to get different relation between sentences' matrix. Nevertheless, we try other filter sizes (for instance 128, 64, 32 or 64, 64, 64), they are not effective.

To tackle to get long-range dependencies as a challenge we add scaled dot-product attention to our network. Although the 1-head attention mechanism has good results, we prefer multi-head attention to achieve results that are more accurate (Tang, et al., 2018).

4. EXPERIMENTAL RESULTS

For evaluating our method, we compare our method's results with those of other solutions. Before we present the results, we introduce our dataset and tools.

4.1. Tools and Dataset

We use several python libraries such as scikit-learn, Keras, TensorFlow, Gensim, and Numpy.

Six books of Nāsir-i Khusraw and eighteen books of other Contemporary authors with a similar style or content create our dataset in Persian literature.

To make our dataset we collect near 4000 pages by other authors from 11, 12, 13, and 14 century AD and 800 pages by Nāsir-i Khusraw (without Khān al-Ikhwān) and randomly select 4000 sentences for training and test (see Table 1).

Selected authors in the dataset:

Nasir-i Khusraw Qubadiyani is the greatest of the Ismaili writers. His works have characteristics as follows:

- First, the works of Nasir-i Khusraw are all written to promote the ideas of the Ismaili sect, in which the thoughts and beliefs of this sect are described and expressed.
- Second, his works are mainly written on theological and philosophical subjects.
- Third, the writings of Nasir-i Khusraw are considered to be scientific works in the Persian language, and they were written in a historical period (fifth century AH) which scientific works in Iran were mostly written in Arabic. Nasir-i Khusraw has placed special emphasis on presenting Persian equivalents for Arabic words and expressions, so his work is a very credible and important source for Persian scientific terms.

The educational data for this study consist of the Persian works of three other authors. Each of these authors and works has characteristics that have made them well-known in the world.

Ibn Sina in his life (359-416 AH) coincides with the youth of Nasir-i Khusraw (394-381 AH) ie the Persian prose of the works of both can be considered to be written in the first period of Persian prose.

Ibn Sina has produced many works on philosophical and theological subjects that are part of these Persian works. Like Nasir-i Khusraw, Ibn Sina was particularly obsessed with using the Persian equivalent of Arabic words and expressions. In general, considering Persian works of Ibn Sina in terms of the type of language (the first period of Persian prose), subject and Persian writing, are similar to the works of Nāsir-i Khusraw.

Muhammad ibn Muhammad ibn al-Hasan al-Tūsī (better known as Nasir al-Din Tūsī):

Born about a hundred years after Nāsir-i Khusraw, and in the period when most of the scientific works in Iran were written in Arabic, like Nāsir-i Khusraw's era.

Table 1. The books used to test methods

No.	Title	Date (Lunar Calendar)	Author	Subject	Category
1	Aghāz Va Anjām	7th Century	Muhammad ibn Muhammad ibn al-Hasan al-Tūsī	Philosophy and Kalām	Theology
2	Asās ul-iqtibās	7th Century	Muhammad ibn Muhammad ibn al-Hasan al-Tūsī	Logic	Natural science
3	Awsaf al-Ashrāf	7th Century	Muhammad ibn Muhammad ibn al-Hasan al-Tūsī	Mysticism	Theology
4	Qurāzeye Tabiʿiyāt	4th Century	Avicenna	Natural science	Natural science
5	Manṭeq-i Dāneš-nāmayi ʿalāʾī	4th Century	Avicenna	Philosophy	Theology
6	Resālayi jūdīya	4th Century	Avicenna		
7	Resāla andar ḥaqīqīyat wa kayfīyat-i selselayi mawjūdāt wa tasalsol-i asbāb wa mosabbabāt (Treatise on reality and the mode of connection of beings and the interconnection of causes and effects)	4th Century	Avicenna	Philosophy and Kalām	Theology
8	Andar dāneš-i rag (Resālayi nabż)	4th Century	Avicenna	Medicine	Natural science
9	Tabiiyat Dāneš-nāmayi ʿalāʾī	4th Century	Avicenna	Natural science	Natural science
10	Tārikh-i Bayhaqī	5th Century	Abul-Fazl Bayhaqi	History	History
11	Tazkirat al-Awliyā	6-7th Century	Aṭṭār	History and Mysticism	Theology
12	History of Sistan	5th Century	Unknown	History	History
13	Tārikh-i Bal'ami	4th Century	Muhammad Bal'ami	History	History
14	Maqsad al-Aqsā	7th Century	Aziz-i Nasafi	Mysticism	Theology
15	Ketāb al-insān al-kāmel	7th Century	Aziz-i Nasafi	Mysticism	Theology
16	Bayān al-tanzil	7th Century	Aziz-i Nasafi	Mysticism	Theology
17	Zobdatal-ḥaqāyeq	7th Century	Aziz-i Nasafi	Mysticism	Theology
18	Kashf al-ḥaqāyiq	7th Century	Aziz-i Nasafi	Mysticism	Theology
19	Safarnama	5th Century	Nāsir-i Khusraw	History	History
20	Khān al-Ikhwan	5th Century	Nāsir-i Khusraw	Philosophy and Kalām	Theology
21	Zād al-Musāfirin	5th Century	Nāsir-i Khusraw	Philosophy and Kalām	Theology
22	Gushayish va Rahayish	5th Century	Nāsir-i Khusraw	Philosophy and Kalām	Theology
23	Wajh-i Din	5th Century	Nāsir-i Khusraw	Philosophy and Kalām	Theology
24	Jami al-Hikmatayn	5th Century	Nāsir-i Khusraw	Philosophy and Kalām	Theology

He wrote important works on philosophy and theology in Persian. Also, Nasir al-Din Tūsī restored the philosophical tradition of Ibn Sina. He was an Isma'ili, lived with the Isma'ilis for about 26 years and wrote many works on their beliefs and thoughts. In general, the structure of Nasir al-Din Tūsī's works and their prose are, for some reason, close to those of Nāsir-i Khusraw.

Aziz-i Nasafi of the most important and influential mystics of the 7th century AH, who lived about two hundred years after Nāsir-i Khusraw. The influence of Isma'ilis on him is evident in his works [2: 8-10].

Aziz-i Nasafi's prose and the structure of his works (especially the Insan al-Kamil) are very similar to those of Nāsir-i Khusraw.For preprocessing books, we first convert them from MS word files to utf-8 txt files then remove footnote, page numbers, and special characters. Some words have different spells in Persian especially in old texts we have to convert them to standard form. We truncate text by threshold and punctuation like dot, question marks, and so on. Books punctuations are edited by special editors as this work is vital. Finally, sentences are chosen randomly for train and test by following methods.

All of the solutions we test use flatten layer at last and the same embedding layer. We only change the middle layers.

4.2. *Training*

We divide processed data into two groups: test data and train data. The test group contains 20% of the data that is chosen randomly. The remaining data, near 6000 sentences, are used as the train data. Converting each word of the sentence to vector by word2vec (as pre-train phase), the embedding layer's data ready to be consumed by our network. whole network train with backpropagation strategy and Adam Optimizer. We set learning rete to 0.01, train iterations 100, and batch size 100.

4.3. *Results*

The results which were shown in Table 2 demonstrate our method's superiority. Furthermore, our solution implies changes in convolution layers and dropout rate how much effect on this kind of classification. Besides, additional dense layers not only decrease classification accuracy but also increase computational overhead. We compare our method with other well-known methods listed in the table.

LSTM: with 100 cells and drop out 0.2 is tested but the results show it has less accurate outputs.

LSTM CNN: We also check CNN combination with LSTM, configuration remains same, but we add a convolution layer (32 filters 3*3) and a maxpooling (2*2) before the LSTM layer this work somewhat improve accuracy.

LSTM CNN2: we use the same configuration as LSTM CNN but use two dense layers.

Kim CNN: a single layer convolution with different filter size contains 3*3, 4*4 and 5*5. With a 0.5 dropout rate.

Our CNN: we apply several configurations for sake of completeness, and the results show the discussed model in the previous section (Our CNN drop out and layer normalization) has the best accuracy among other solutions.

Table 2 Comparison with other methods. Accuracy is between two class Belong to Nasir-i Khusraw or not.

Methods	Test accuracy
Our CNN same drop out (0.65)	70.55
Our CNN drop out and layer normalization with self-attention (our configuration)	72.59%
3 CNN different filter size(Kim CNN)	70.21%
Our CNN with two dense layers	66.61%
LSTM	70.78%
LSTM_CNN	70.91%
LSTM_CNN2	66.31%
Linear SVM	9.23%
Sigmoied SVM	13.03%
Polynomial SVM	17.72%
Kmeans	14.98
Gussian Naïve bayes	12.67%

Some other methods like SVMs, Kmeans, and Gussian Naïve Bayes were used for this classification, but their results were very poor.

Results show Khan al-Ikhawan belongs to Nāsir-i Khusraw with 86.42% present accuracy. After training our model, we tested Khan al-Ikhawan text with our network and the results demonstrated by our system approved this.

Another note is training the embedding layer separately to be more effective with 150 vector length (some research works useed 300 vector length but in our work its effects is not considerable). We also trained our embedding layer with both Nāsir-i Khusraw and other authors because we want our system to distinguish between texts without the help of embedding layers. We concluded that training each author text separately improves classification performance up to 3%.

5. Conclusion

The objective of this article was Authorship Attribution by using Deep learning. By using novel CNN architecture, we confirmed the true author of an old Persian book. Furthermore, our method showed improvements against other state-of-art methods. Also, our proposed method does not depend on any specific language or special text and it is a context-free solution.

Thanks to deep learning now we can respond to enigmatic questions unanswered by researchers for many centuries. As a case study, we identified the author of Khan al-Ikhwan as Nāsir-i Khusraw by deep learning.

For future works, we suggest using Morkov Random Fields with CNN to better inference in such texts and Generalized Autoregressive Pretraining formulation to get better results.

References

Alharthi, H., Inkpen, D., & Szpakowicz, S. (2018). Authorship Identification for Literary Book Recommendations. In Proceedings of the 27th International Conference on Computational Linguistics, Santa Fe.

Amancio, D. R. (2015). Comparing the topological properties of real and artificially generated scientific manuscripts. Scientometrics, 105(3), 763-1779.

Amancio, D. R. (2015). Probing the Topological Properties of Complex Networks Modeling Short Written Texts. PLOS ONE, 10(2), pp. 1-17.

Badrinarayanan, V., Kendall, A., & Cipolla, R. (2017). SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(12), 2481-2495.

Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural Machine Translation by Jointly Learning to Align and Translate. CoRR, abs/1409.0473.

Chen, M. (2017). Efficient vector representation for documents through corruption. arXiv preprint arXiv:1707.02377.

Chollet, F. (2017). Xception: Deep Learning with Depthwise Separable Convolutions. Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu.

Cutler, A. 1983. Speakers' Conception of the Functions of Prosody. In: A. Cutler & R. D. Ladd, eds. Prosody: Models and Measurements. Berlin: Springer-Verlag, pp. 79-92.

Diurdeva, P., Mikhailova, E., & Shalymov, D. (2016). Writer identification based on letter frequency distribution. 19th Conference of Open Innovations Association (FRUCT), Jyvaskyla.

Fathullah, M. Khorasani, S. & Jafari N. S. Ibn Sina, The Great Islamic Encyclopedia Volume 4. Tehran: Cetor Of Great Islamic Encyclopedia.

Hitschler, J., van den Berg, E., & Rehbein, I. (2017). Authorship Attribution with Convolutional Neural Networks and POS-Eliding. In Proceedings of the Workshop on Stylistic Variation, Copenhagen.

Hosseini, M. (2001). Introduction to Hadiqat al-Hagiqah and Sharia al-Tareqah. Tehran: Tehran University publishing center.

Howard, A., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. ArXiv, abs/1704.04861.

Ioffe, S., & Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Proceedings of the 32Nd International Conference on International Conference on Machine Learning, Lille.

Ja'far Shahidi, S. (1994). Yesterday Until Today (Articles Collection). Tehran: Tehran Ghatreh.

Kalchbrenner, N., Grefenstette, E., & Blunsom, P. (2014). A convolutional neural network for modelling sentences. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, Baltimore.

Kestemont, M., Stover, J., Koppel, M., Karsdorp, F., & Daelemans, W. (2016). Authenticating the writings of Julius Caesar. Expert Systems with Applications, 63, 86 - 96.

Kim, Y. (2014). Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882.

Le, Q., & Mikolov, T. (2014). Distributed Representations of Sentences and Documents. In Proceedings of the 31st International Conference on International Conference on Machine Learning, Beijing.

Li, C. Su, Y. Lin, Tsai, C. Chang, W. Huang, & K. Juan, Y. (2015). Combination of Feature Engineering and Ranking Models for Paper-author Identification in KDD Cup 2013. J. Mach. Learn. Res., 16(1), 2921-2947.

Luong, M. Pham, H., & D. Manning, C. (2015). Effective Approaches to Attention-based Neural Machine Translation. CoRR, abs/1508.04025. Availabe at: http://arxiv.org/abs/1508.04025

Ma, X., & Hovy, E. (2016). End-to-end sequence labeling via bi-directional lstm-cnns-crf. arXiv preprint arXiv:1603.01354.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

Minavi, M. (2001). The old manuscripts should be the criterion for correcting literary texts. Ayeneh Miras, (15), 89 - 97.

Mirbaqerifard, A.,2000, Bayan al Tanzil, Correction and Introduction by Sayyed Ali Asghar Mirbaqeri Fard, Tehran: Anjomane Mafakhere Melli.

Mnih, V., Heess, N., Graves, A., & Kavukcuoglu, K. (2014). Recurrent models of visual attention. arXiv preprint arXiv:1406.6247.

Moein, M., 1956, "Nasir al-Din Tūsī and Persian Language and Literature", Journal of Faculty of Literature and Humanities, University of Tehran, 4, 33.

Pennington, J., Socher, R., & Manning, C. D. (2014). Glove: Global vectors for word representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, Doha.

Savoy, J. (2012). Authorship Attribution Based on Specific Vocabulary. ACM Trans. Inf. Syst., 30(2), 1046-8188.

Sboev, A., Litvinova, T., Gudovskikh, D., Rybka, R., & Moloshnikov, I. (2016). Machine learning models of text categorization by author gender using topic-independent features. Procedia Computer Science, 101, 135-142.

Seroussi, Y., Zukerman, I., & Bohnert, F. (2014). Authorship Attribution with Topic Models. Comput. Linguist., 40(2), 269-310.

Shalymov, D., Granichin, O., Klebanov, L., & Volkovich, Z. (2016). Literary writing style recognition via a minimal spanning tree-based approach. Expert Systems with Applications, 61, 145 - 153.

Shrestha, P. a., Montes, M., Rosso, P., & Solorio, T. (2017). Convolutional neural networks for authorship attribution of short texts. In roceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, Valencia.

Srivastava, N. a., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. J. Mach. Learn. Res., 15(1), 1532-4435.

Tang, G., M ̈uller, M., Rios, A., & Sennrich, R. (2018). Why Self-Attention? A Targeted Evaluation of Neural Machine Translation Architectures. Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., & Gomez, A. Polosukhin, I. (2017). Attention is all you need. NIPS 2017.

Velásquez, J., & Oberreuter, G. (2013). Text mining applied to plagiarism detection: The use of words for detecting deviations in the writing style. Expert Systems with Applications, 40(9), 3756 - 3763.

Wang, S., & Manning, C. (2013). Fast dropout training. Proceedings of the 30th International Conference on Machine Learning, Atlanta.

Wu, F., Fan, A., Baevski, A., Dauphin, Y., & Auli, M. (2019). Pay Less Attention with Lightweight and Dynamic Convolutions. CoRR, abs/1901.10430. Retrieved from http://arxiv.org/abs/1901.10430.

Yin, W., Schutze, H., Xiang, B. & Zhou, B., (2016). ABCNN: Attention-Based Convolutional Neural Network for Modeling Sentence Pairs. Transactions of the Association for Computational Linguistics, Volume 4, p. 259–272.