Authorship Verification of Literary Texts with Rhythm Features

Ksenia Lagutina, Nadezhda Lagutina
P.G. Demidov
Yaroslavl State University
Yaroslavl, Russia
ksenia.lagutina@fruct.org,
lagutinans@gmail.com

Elena Boychuk
Yaroslavl State Pedagogical
University named after K.D.Ushinsky
Yaroslavl, Russia
elena-boychouk@rambler.ru

Vladislav Larionov, Ilya Paramonov
P.G. Demidov
Yaroslavl State University
Yaroslavl, Russia
vladlarionov998@gmail.com,
ilya.paramonov@fruct.org

***Abstract*—The paper is devoted to the authorship verification of literary texts of 19th–21st centuries using rhythm features and statistical analysis of these features. The authors presented algorithms that fully automatically find lexico-grammatical figures in raw texts. Then the authors calculated rhythm features basing on the frequency of the appearance and the structure of these figures. The experiments showed that many English, Russian, French, and Spanish authors were successfully verified with their rhythm features, the best F-measure for the AdaBoost classifier achieved 88–96 %. Besides, rhythm features were clearly visualized in boxplots and heatmaps that allowed to compare the text rhythm in a whole for several authors and languages simultaneously.**

I. Introduction

An important task in humanitarian and forensic research is to check whether the text belongs to a given author or not. A significant part of the work in this area is devoted to the verification of the authors of emails, messages in messengers and social networks [1], [2]. The authorship verification for scientific and journal articles is an integral part of the resolution of copyright disputes [3], [4]. In addition to research on contemporary texts, scientists are interested in the verification of ancient authors [5], [6], Renaissance authors [7], 18th century writers [8].

Although the methods and the tools for the automatic verification of the authorship vary considerably, almost all researchers use the author's style features. The most popular ones are easily computable text features, such as word unigrams and n-grams of the characters. But in recent years, more and more authors pay attention to the peculiarities of vocabulary, grammar, idioms, and phonetics of the author's text —all that in classical linguistics is called the idiolect [9].

Among all the features of the author's style, there is almost no works devoted to the verification of the authorship use rhythm figures of speech based on the repetition of words and phrases, unlike other stylistic features (n-grams, syntactic structure, punctuation) that are successfully applied in many research [10]. Nevertheless, the rhythm features are used to analyze the works of art in philology. There is the evidence that they are useful for the analysis of the author's style [11]. We successfully apply these features to differentiate the works by centuries [12].

This paper extend our investigations of the rhythm features. The goal of this research is to verify the authors of the artistic prose basing on rhythm features only. From the results of authorship verification we can conclude how good rhythm features separate the author from others and and how the author's texts are homogeneous in rhythm. The subtask of this investigation is the comparison of the style of particular authors based on the visualization of rhythm statistics.

The paper is structured as follows. Section II the describes state-of-the-art research in authorship verification. In Section III we present new algorithms for the rhythm figures extraction. Section IV describes the design of the experiments with computation and visualization of statistical features of the text rhythm and authorship verification. In Section V we visualize and compare the style of particular authors. Section VI describes the experiments with the authorship verification. In Section VII we analyze and interpret the authorship verification from computer science and linguistics points of view. Conclusion summarizes the paper.

II. State of the art

The text features that are used for natural language processing, are classified into low-level features and high-level or linguistic ones. Low-level features include character-based and word-based features like embeddings, simple statistics, token-based features, etc. Linguistic features include syntactic (based on punctuation, syntactic structure, etc.), semantical, stylistic, and many others [10]. Rhythm features are stylistic features that are based on the repetition of language units (words, phrases, types of sentences, etc.) [13].

Use of various feature types for the the automatic attribution of the documents was investigated by many scientists [14].

Lee et al. [1] developed algorithms and researched different classifiers to determine the authenticity of short messages on social networks (the average length is 20.6 words) from Facebook. The authors used 233 features, including 227 stylometric ones, such as character-level: numbers of alphabets, capital letters, special characters; word-level: total word count, average word length, word count with 1 character, etc.; syntactic: number of punctuation marks and functional words, total number of sentences, and six social media-specific features, including emoticons, abbreviations, beginning of a sentence without capital letter, ending of a sentence without a punctuation mark. The results of the experiments showed an average accuracy of 79.6 % for 30 users and 9259 messages. This quality was achieved due to the stylometric features. The social media features did not improve the classification.

The other researchers achieved good results using n-grams. Brocardo et al. [15] proposed a supervised learning method combined with n-gram analysis to verify the authorship of short texts taken from the Twitter and Enron email corpus. The average word count per email was 200. The emails were plain texts and covered a variety of topics, from business communications to technical reports and face-to-face chats. They conducted an experimental evaluation of texts of the 87 authors that gave the results consisting of an error rate from 5.48 % to 12.3 %.

An example of classical research in the field of the authorship verification of scientific and news articles using stylometric parameters is based on n-grams [16]. The verification is realized by means of the determination of the proximity of the numerical feature vectors of the documents. The method was applied to five languages: Dutch, English, Greek, Spanish, and German. The F-measure varied from 67.37 % for Greek up to 83.33 % for Spanish. The authors' method also showed good results at the PAN-2020 competition [4], [14].

To verify the authorship of the articles in Arabic, Ahmed [17] used lexical, morphological, and syntactic features and feature ensembles. The experiments on a quite small corpus of 31 books showed the accuracy of 87 %. The analysis of the efficiency of different types of features showed the advantage of applying features based on the syntactic structures of the text.

The method of Adamovic et al. [18] showed a high result over 90 % of the accuracy. The authors identified language-independent text stylistic features and used the SVM-RFE (Support Vector Machine based on Recursive Feature Elimination) feature selection method to remove redundant and irrelevant characteristics from the learning process. The method was applied to the verification of the authorship of articles in four languages: English, Greek, Spanish, and German.

Boenninghoff et al. [19] pointed out that the reliability of using standard stylometric features in machine learning algorithms significantly decreased for short and thematically diverse texts on social networks. To verify the authorship of short Amazon reviews, the researchers used Siamese neural networks to visualize decision-making. The character-level features were used to construct a feature vector, but when discussing the results, the authors carried out a linguistic analysis of the internal weights of the network in order to interpret the result from the point of view of traditional linguistic categories. It should be noted that this work uses a large corpus of texts 9 052 606 reviews written by 784 649 authors, which volume, of course, improves the quality of the problem solution.

The task of the authorship verification of short texts available in a small volume was solved in [20] by building a language text model. In this paper, the researchers considered the authorship verification problem for unauthorized malicious publications on social networks. The corpus included the texts by 103 authors, at least 300 texts for each author. The F-measure was 74 %.

Many researchers raised the authorship verification quality by improving and combining the classification methods. Boenninghoff et al. [21] proposed a new neural network topology to answer the question whether two documents with unknown authors were written by the same author or not. This approach performed better for short multi-genre social media posts than the algorithms based on traditional linguistic features such as n-grams. Precision, recall, and F-measure reached 84 %.

To verify the authorship of short articles in English, Benzebouchi et al. [22] proposed a machine learning model scheme based on a combination of three different architectures: convolutional neural networks, recurrent convolutional neural networks, and machine support vector classifiers. The final decision was obtained by combining the results of three models using the voting method. Word2vec was used as a text model. As a result of experiments, the accuracy was from 91 % to 97 %. Unfortunately, other quality indicators, such as precision and recall, were not indicated in the paper.

In the computational linguistics research, the authorship attribution or verification of literary texts in most cases was solved using predominantly character-level, word-level or syntactic features [10]. Other linguistic features were usually applied as a part of a complex text model and almost did not investigated separately.

Nevertheless, several studies use exclusively stylistic features of the text. For example, it is so for deciding on the authenticity of Pliny the Younger's letter to Trajan concerning the Christians [6], the assessment of whether the controversial work "The Epistle to Cangrande" was written by Dante Alighieri [7], the authorship verification of Johann Wolfgang Goethe's anonymous contributions to the journal "Frankfurter gelehrte Anzeigen" [8]. These researchers considered stylometric features based on specific phrases that charactered the author or the time of writing. The use of these features allowed to perform an additional analysis of the relevance of the obtained results. The quality of the authorship verification achieved 88–98 % but only on the small set of 5–6 authors. All authors emphasized the ambiguity of the results and the need to continue research. The main direction of these studies is associated primarily with complex stylistic features.

Thus, the stylistic features play an important role in the authorship verification. However, no automatic system uses the rhythm features as a text model. On the one hand, this is due to the relative rarity of these figures and the fact that they are associated primarily with poetic works. On the other hand, the existing text processing libraries make it possible to effectively find complex morphological and syntactic properties of a text, for example, the syntactic role of a word in a sentence and its functional relationship with other words. Using rhythm features requires implementation of new search algorithms, so the researchers cannot add these parameters to the classification system quickly. Therefore, the main questions that appear in this field of natural language processing are how the rhythm features characterize the author's style and whether they are applicable to the authorship verification. In this article we are trying to answer these questions.

III. Algorithms for rhythm figure search

A. Rhythm figures used in research

Our task is to extract rhythm features from literary texts, visualize them and apply for authorship verification to analyze how these features can distinguish authors and what authors have the unique homogeneous rhythm in many texts.

We study rhythm features that are based on the lexico-grammatical rhythm figures:

- *anaphora*, a repetition of sequence of words at the beginning of neighboring sentences. For example, "**I wanted** a miracle job advertisement. **I wanted** someone to come along and say";
- *anadiplosis*, a repetition of the same word at the end of a clause and at the beginning of the following clause. For example, "It was right to do **it, it** was kind to do **it, it** was benevolent to do it, and he would do it again";
- *diacope*, a repetition of a word or phrase with intervening words within one sentence. For example, "**Help**, Charmian, **help**, Iras";
- *epanalepsis*, a repetition of the initial part of a sentence at the end of the same sentence. For example, "**The king** is dead, long live **the king**";
- *epiphora*, a repetition of the same word or words at the end of neighboring sentences (also called epistrophe). For example, "Frank **knew**. And Maxim did not know that he **knew**";
- *epizeuxis*, a repetition of a word or phrase in immediate succession within one sentence. For example, "**Weak**! **Weak**! **Weak**!";
- *polysyndeton*, a repetition of the same conjunction within one sentence (simple and pair conjunctions and conjunctive adverbs can be repeated). For example, "There were frowzy fields, **and** cow-houses, **and** dunghills, **and** dustheaps, **and** ditches";
- *symploce*, a repetition of the beginning and the end of two or more neighboring sentences, combination of anaphora and epiphora. For example, "**I'm** wanting **to tell you**. **I'm** waiting **to tell you**".

Search algorithms for these figures and their evaluation were described in our previous work [23].

In this research the list of figures is extended by the following rhythm figures:

- *aposiopesis*, a figure of speech in a sentence which is deliberately broken off and left unfinished. For example, "She resurrected nothing but the cat ... but the cat ...";
- *repeating interrogative sentences*, a repetition of the interrogative point at the ending of neighboring sentences. For example, "Where's my car? Where's my house?";
- *repeating exclamation sentences*, a repetition of the exclamation point at the ending of neighboring sentences. For example, "Jeepers! You scared the life out of me!";
- *chiasmus*, a reversal of grammatical structures in successive phrases or clauses with the repetition of words. For example, "**You forget** what **you** want to **remember**, and **you remember** what **you** want to **forget**".

All the given algorithms work with a text previously split into sentences, which are, in turn, split into words. In addition, to search a figure, the algorithms use stop words specific for every figure. The lists of stop words were formed manually by experts. They include prepositions, articles, functional words, pronouns, and auxiliary verbs.

As a result, all the algorithms produce the lists of figures, every figure specified with the words from a text that form the figure, and a context—a sentence or sentences in which the figure has occurred.

The details of each algorithm including their implementations in pseudo-code are provided below.

B. Aposiopesis searching algorithm

The algorithm looks at the punctuation mark at the end of each sentence in the given list. If the punctuation mark is an ellipsis, then the algorithm saves the position of the first sentence word. After that the algorithm steps through the next sentences and switches on the *in_repetition* flag until the sentence has an ellipsis as the punctuation mark at the end of the sentence. As soon as the sentence ends with no ellipsis and the *in_repetition* flag is true, the algorithm adds an aposiopesis with the context range from $feature\_start$ till $word\_count - 1$ to the list of aposiopeses.

```
Require: sentences as list $S$
  $A := \varnothing$
  $word\_count := 0$
  $feature\_start := None$
  $in\_repetition :=$ false
  for $sentence$ in $S$ do
    if $sentence.ending\_punct = $ "..." then
      if $feature\_start = None$ then
        $feature\_start := word\_count$
      else
        $in\_repetition :=$ true
      end if
    else
      if $in\_repetition$ then
        append aposiopesis(context = [feature_start,
        word_count - 1]) to $A$
        $feature\_start := None$
        $in\_repetition :=$ false
      end if
    end if
    $word\_count := word\_count + len(sentence)$
  end for
  if $in\_repetition$ then
    append aposiopesis(context = [feature_start,
    word_count - 1]) to $A$
    $feature\_start := None$
    $in\_repetition :=$ **false**
  **end if**
**Ensure:** list of aposiopeses $A$
```

C. Algorithm for searching repeating interrogative sentences

The algorithm is similar to the aposiopesis searching algorithm. The algorithm steps by sentence in the given sentence list and, for each of them, looks at the punctuation mark at the end of the sentence. If the punctuation mark is in {”?”,”?!”,”?...”,”??”,”???”}, then the algorithm saves the position of the first sentence word. After that the algorithm steps through the next sentences and switches on the *in_repetition* flag until the sentence has the same punctuation mark at the end of the sentence. As soon as the sentence ends with other punctuation mark and the *in_repetition* flag is true, the algorithm adds a feature with the context range from $feature\_start$ till $word\_count - 1$ to the list of repeating interrogative sentences.

```
**Require:** sentences as list $S$
  $I := \varnothing$
  $word\_count := 0$
  $punct\_list := \{”?”,”?!”,”?...”,”??”,”???”\}$
  $feature\_start := None$
  $in\_repetition :=$ **false**
  **for** $sentence$ **in** $S$ **do**
    **if** $sentence.ending\_punct$ **in** $punct\_list$ **then**
      **if** $feature\_start = None$ **then**
        $feature\_start := word\_count$
      **else**
        $in\_repetition :=$ **true**
      **end if**
    **else**
      **if** $in\_repetition$ **then**
        append *interrogative_sentences(context = [feature_start, word_count - 1])* to $I$
        $feature\_start := None$
        $in\_repetition :=$ **false**
      **end if**
    **end if**
    $word\_count := word\_count + len(sentence)$
  **end for**
  **if** $in\_repetition$ **then**
    append *interrogative_sentences(context = [feature_start, word_count - 1])* to $I$
    $feature\_start := None$
    $in\_repetition :=$ **false**
  **end if**
**Ensure:** list of repeating interrogative sentences $I$
```

D. Algorithm for searching repeating exclamatory sentences

The algorithm is similar to the algorithm for the searching of repeating interrogative sentences. The main difference is the the punctuation mark multitude. The algorithm steps by sentence in the given sentence list and, for each of them, looks at the punctuation mark at the end of the sentence. If the punctuation mark is in {”!”,”?!”,”!...”,”!!”,”!!!”}, then the algorithm saves the position of the first sentence word. After that the algorithm steps through the next sentences and switches on the *in_repetition* flag until the sentence has the same punctuation mark at the end of the sentence. As soon as the sentence ends with other punctuation mark and the *in_repetition* flag is true, the algorithm adds a feature with the context range from $feature\_start$ till $word\_count - 1$ to the list of repeating exclamation sentences.

```
**Require:** sentences as list $S$
  $E := \varnothing$
  $word\_count := 0$
  $punct\_list := \{”!”,”?!”,”!...”,”!!”,”!!!”\}$
  $feature\_start := None$
  $in\_repetition :=$ **false**
  **for** $sentence$ **in** $S$ **do**
    **if** $sentence.ending\_punct$ **in** $punct\_list$ **then**
      **if** $feature\_start = None$ **then**
        $feature\_start := word\_count$
      **else**
        $in\_repetition :=$ **true**
      **end if**
    **else**
      **if** $in\_repetition$ **then**
        append *exclamatory_sentences(context = [feature_start, word_count - 1])* to $E$
        $feature\_start := None$
        $in\_repetition :=$ **false**
      **end if**
    **end if**
    $word\_count := word\_count + len(sentence)$
  **end for**
  **if** $in\_repetition$ **then**
    append *exclamatory_sentences(context = [feature_start, word_count - 1])* to $E$
    $feature\_start := None$
    $in\_repetition :=$ **false**
  **end if**
**Ensure:** list of repeating exclamation sentences $E$
```

E. Chiasmus searching algorithm

The algorithm steps by pairs of sentences from the given sentence list. In each pair the algorithm checks whether the first word of the first sentence in the pair equals the last word of the second sentence in the pair and whether the last word of the first sentence in the pair equals the first word of the second sentence in the pair. If they do, the algorithm has found a new chiasmus. Then the algorithm adds to the chiasmus word list the first and the last words of the sentences of the pair and assigns the range from the first word of the first sentence in the pair till the last word of the second sentence in the pair as the context of the chiasmus.

```
**Require:** sentences as list $S$
  $C := \varnothing$
  $word\_count := 0$
  **for** $i := 1, \ldots, len(S) - 1$ **do**
```

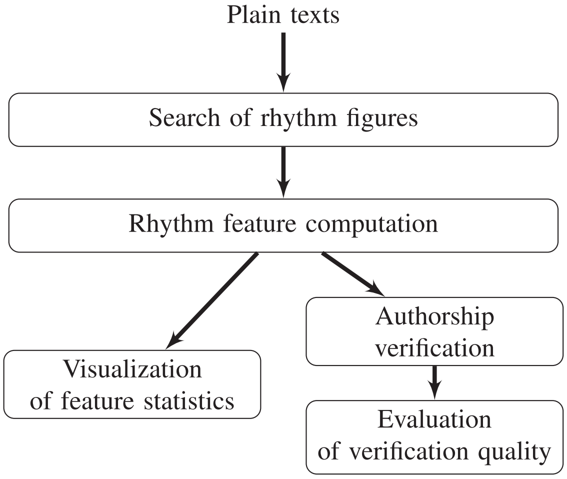

Plain texts
Search of rhythm figures
Rhythm feature computation
Authorship verification
Visualization of feature statistics
Evaluation of verification quality

Fig. 1. Structure of experiments

```
    $sentence := sentences[i]$
    $next\_sentence := sentences[i+1]$
    if $sentence.first\_word = next\_sentence.last\_word$ and
    $sentence.last\_word = next\_sentence.first\_word$ then
      append chiasmus(context = [
          sentence.first_word,
          next_sentence.last_word
      ],
      words = [
          sentence.first_word,
          sentence.last_word,
          next_sentence.first_word,
          next_sentence.last_word
      ]) to $C$
    end if
    $word\_count := word\_count + len(sentence)$
  end for
Ensure: list of chiasmuses $C$
```

The precision of the search algorithms was computed by experts in classical linguistics manually. Four researchers processed a total of 24 texts of different authors, randomly selected from the corpus. Each expert worked 16 hours. She manually evaluated precision of search for all rhythm figures. The methodology of expert analysis was described in more detail in our paper [24].

The precision of the figure search reached 80–95 %. The use of these algorithms allow to represent an author style of texts in terms of rhythm figures.

IV. Design of experiments

A. Overview

The structure of the experiments with text rhythm includes the main stages visualized in Fig. 1. Firstly, we find rhythm figures using algorithms from Section III. Secondly, we compute the statistical features based on the rhythm figures. Then, we use the features in two ways: visualize them to perform the statistical analysis and collect them in vectors for the authorship verification. Finally, the verification results are estimated using the standard quality measures.

Let us discuss these stages in more detail.

B. Computation of statistical rhythm features

We compute the following statistical rhythm features, separately for each text:

- features for visualization:
  - the number of all lexico-grammatical figures divided by the number of sentences and multiplied by 100—the average number of figures per 100 sentences;
  - the percentages of figures among all figures;
- features for verification:
  - the number of occurrences of a figure (anaphora, epiphora, etc.) in a text divided by the number of sentences;
  - the fraction of the unique words—words that repeat only once in rhythm figures;
  - the fractions of nouns, verbs, adverbs, and adjectives—among all the words that appear among rhythm figures.

The first four types of features are counted for all rhythm figures. The computation of the last two types requires only figures based on content words repetitions—lexical ones.

So to each text we assign a vector of statistical features. Several features describe the rhythm figures as independent units, other features represent a structure of figures. For visualization we take the average number of features per 100 sentences and the percentages of features, because they are the most demonstrative.

C. Means of visualization

For the visualization we form a table where the rows correspond to texts of known authors, the columns—to 18 rhythm features. For each author we compute a vector by selecting rows of this author and calculating average over each feature. These vectors are visualized in two ways:

- As boxplots with feature values multiplied by 100 on the x axis and authors on the y axis. The boxplot shows the first and the third quartiles of values as a box, the sample median as a vertical line inside the box, the minimum and the maximum values as ends of the horizontal line, the white circle as the average, and rhombuses as outliers.
- As heat maps that describe ranges of the feature values. The features are located on the x axis and the authors are on the y axis. The map cells contain the feature value, and a tint that indicates the value relative to the others. The smallest values are indicated by lighter tints, the largest values are indicated by dark ones. A bar with a range and tints for different values is displayed on the right of the map.

Both visualization methods are quite clear and allow to analyze homogeneity of the author's style, frequency of feature use by each author, and the difference in author's rhythm statistics.

D. Design of authorship verification

We perform the authorship verification as a binary classification. For each author we divide the texts into two classes: belonging or not belonging to the author. The classifiers are the following supervised algorithms that are proven to be reliable in text classification [12], [25]:

- AdaBoost classifier—a machine learning algorithm that combines the results of 50 Decision Tree classifiers adjusting incorrectly classified texts;
- Random Forest classifier—a machine learning algorithm that averages the results of 50 Decision Tree classifiers;
- Bidirectional LSTM—a recurrent neural network with a Bidirectional Long Short Term Memory (LSTM) layer with 64 units and a dense output layer that uses the sigmoid activation function. The loss function is categorical cross-entropy, the optimization algorithm is Adam, the number of epochs is 100.

For all classifiers we apply the five-fold cross-validation technique: 80 % of texts are the training samples, 20 % are the test ones. The predictions of the binary classification are evaluated with three standard measures: precision, recall, and F-score [26].

All the described figure search algorithms and utilities for author's style visualization and authorship verification are published as parts of the ProseRhythmDetector tool, which is available on the Internet at https://github.com/text-processing/prose-rhythm-detector. It is written in Python programming language and uses Stanza 1.1.1 NLP library for text representation and determination of parts of speech. For the visualization it uses Seaborn 0.11.0 and Matplotlib 3.3.2. For the verification it applies Scikit-Learn 0.23.2 and Keras 2.4.3.

E. Text corpora

We experiment with the text corpora for four languages: English, Russian, French, and Spanish. We created the corpora for this research manually collecting literary works of famous authors.

Each of the English, Russian, and French corpora contains 800 texts of 20 famous authors of 19th–21st centuries, 40 texts per author. The Spanish corpus has 320 texts of 8 authors of 19-th–the beginning of the 20th century.

All the texts represent the fragments of literary works written by the authors in their native language. Each text has the size about 50 000 characters including spaces. So these texts are equal in volume.

V. Author style visualization

For preliminary analysis, we evaluated the suitability of the rhythm features for the authorship verification.

For each text corpus we visualize how frequently lexico-grammatical figures are used by the authors and what figures are the most popular.

To analyze the frequency of all the features we show the distribution of the average number of rhythm figures per 100 sentences—density of figures—in Fig. 2. The boxplot in each subfigure illustrates the language.

The majority of English authors have about 50–100 rhythm figures per 100 sentences. Scott, Kingsley, Eliot, and Henty use the figures more frequently than the others: up to 200–250. Kingsley and Henty also significantly vary the figures' density: they have the largest difference between the maximum and minimum values. Pratchett, Bindloss, McEwan, and Moyes have the least average densities less than 50 figures per 100 sentences and also the smallest ranges of feature values. Pratchett and Hardy have the greatest numbers of outliers: five and four correspondingly, whilst the most of the others have one or two, or does not have such feature values at all.

The most of Russian authors also have about 50–100 rhythm figures per 100 sentences in average, but not more. The authors with the least numbers of figures are Pikul', Pelevin, Vodolazkin, and Prohanov. Pikul', Vodolazkin, Prohanov, and Makanin has the smallest ranges of features. But among Makanin's texts there are eight outliers. Among Rubanov's texts six are outliers, although their density varies from 25 to 250 rhythm figures per 100 sentences. Other authors with high variability of rhythm density are Lev Tolstoy and Leskov.

The texts of French authors have the less density of rhythm figures: 40–60 per 100 sentences. And the variability of this feature is not as significant as for other languages. The most of French authors have the similar average, median, and range of values. The exceptions are Verne, Maupassant, Proust, and Pancol. Verne, Maupassant, and Proust have larger average values than the others, Maupassant and Proust have the greatest variability of density, the texts by Pancol contain outliers with very large values about 150 and 300.

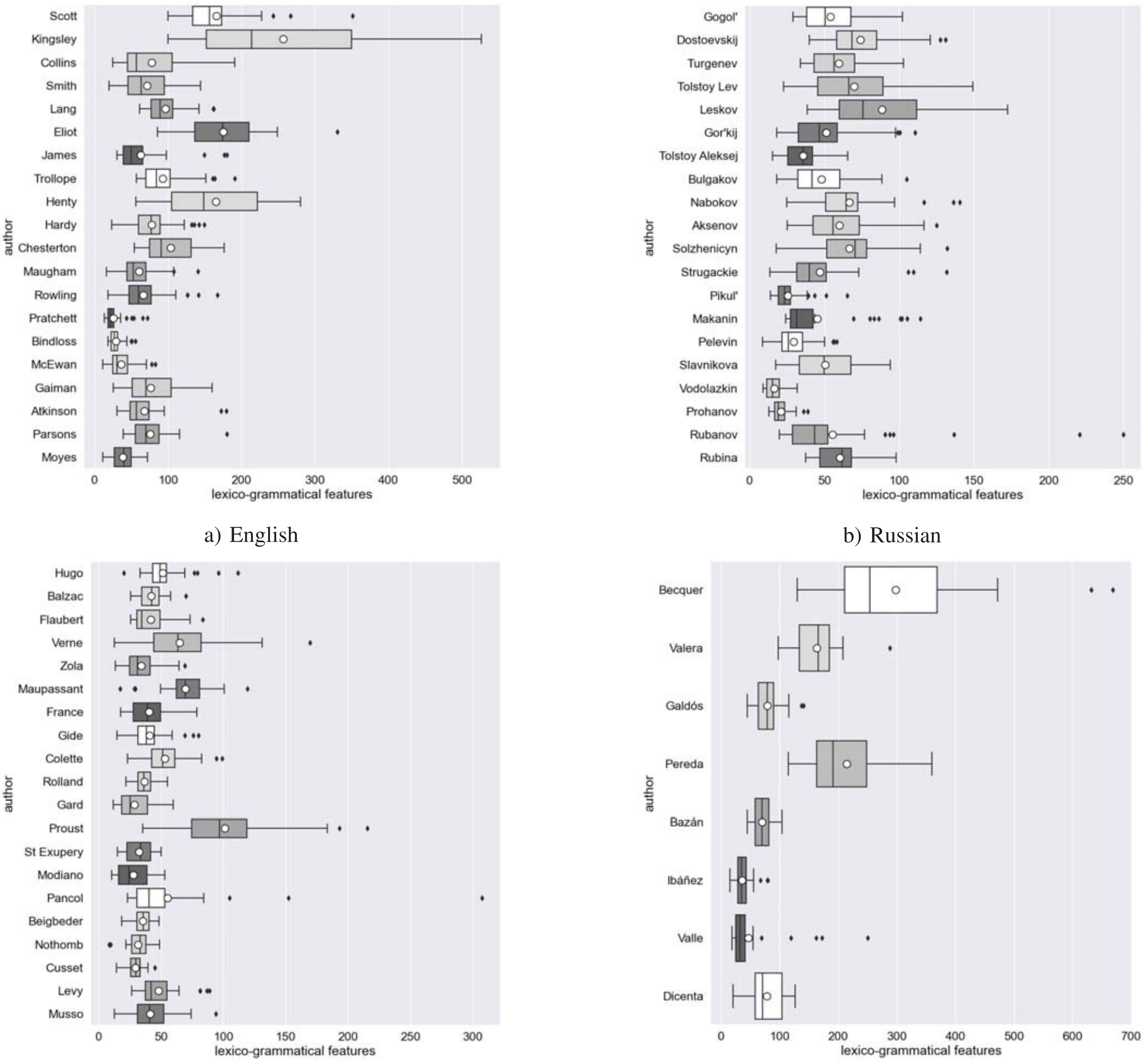
The Spanish authors also contain several authors who are quite similar to each other: the average density from 50 to 100 and the quite small range of the distribution. Becquer and Pereda stand out significantly. They have the large difference between the maximum and minimum values and the highest averages. Becquer's texts also contain two outliers with great rhythm density: 600–700 figures per 100 sentences.

Moreover, we compare the most popular features. They are diacope, polysyndeton, anaphora, and epiphora, because they have the highest percentages among all lexico-grammatical features. These percentages are visualized in heatmaps (Fig. 3).

We can see that the most frequent feature for all languages and authors is diacope, the second one is polysyndeton, the third one is anaphora. Other features appear significantly rare that is illustrated by epiphora.

English authors have 56–73 % of diacope and 13–26 % of polysyndeton. Only Bindloss' texts have 56 % of diacope, in other texts this feature is more than 61 % in average. Pratchett uses epiphora (6.7 %) almost as often as anaphora (7.4 %).

Russian authors have 47–66 % of diacope and 12–32 % of polysyndeton. It is the least range for diacope and the highest range for polysyndeton among all languages. The range of the anaphora percentage is less than in English, but more than in French and Spanish. Among all authors Prohanov, Vodolazkin, and Makanin stand out. Prohanov and Makanin have the least percentages of diacope and the highest percentages of



a) English

b) Russian

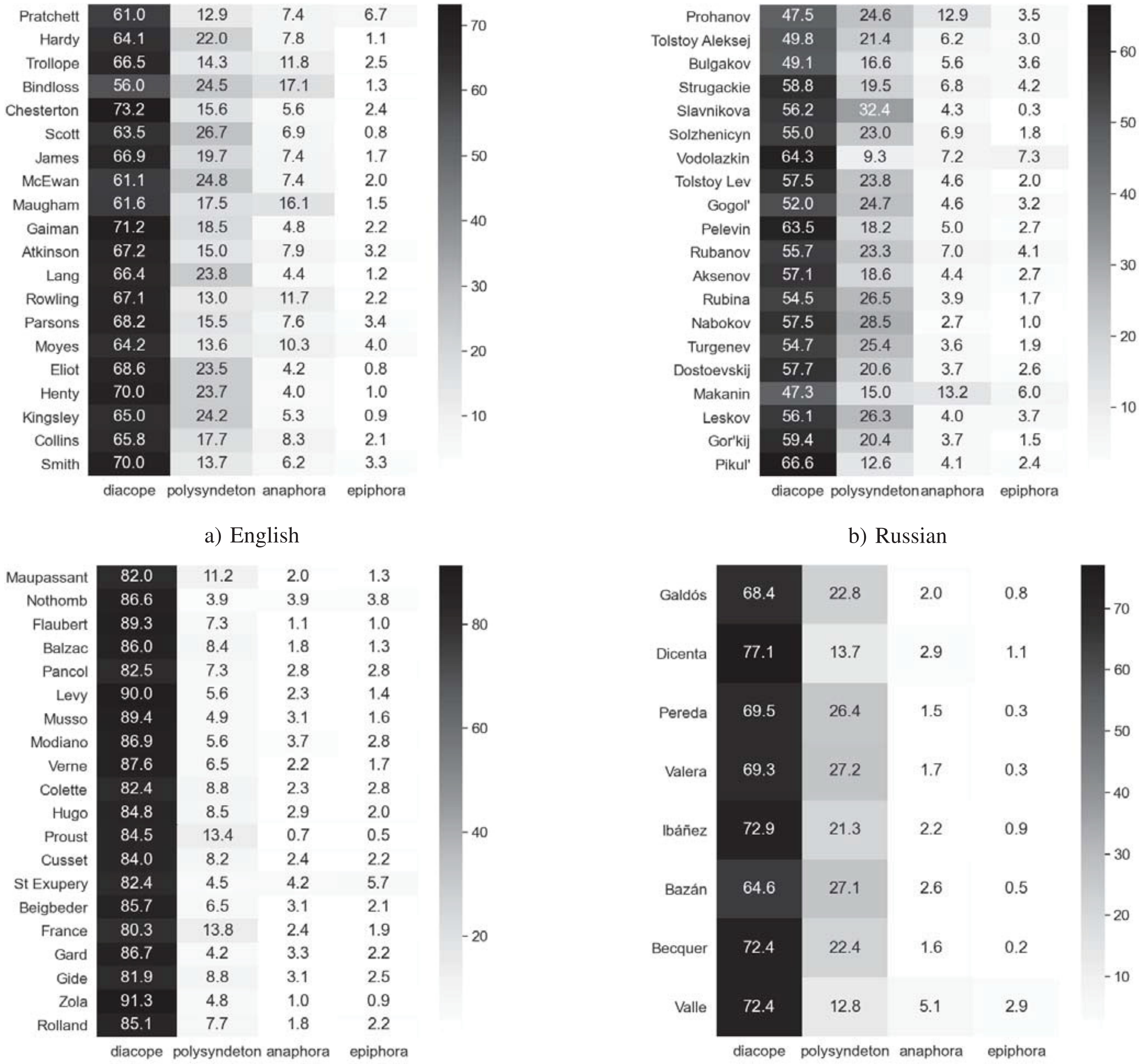
c) French

d) Spanish

Fig. 2. Boxplots with lexico-grammatical figures of authors

TABLE I. COMPARISON OF CLASSIFIERS

Classifier	Language	Precision	Recall	F-measure
AdaBoost	English	**82.0**	**75.2**	**78.5**
RandomForest	English	61.8	55.1	58.3
LSTM	English	69.7	64.5	67.0
AdaBoost	Russian	**85.7**	**76.2**	**80.7**
RandomForest	Russian	65.8	57.6	61.4
LSTM	Russian	73.2	67.1	70.0
AdaBoost	French	**84.5**	**74.4**	**79.1**
RandomForest	French	61.1	53.7	57.2
LSTM	French	67.8	61.1	64.3
AdaBoost	Spanish	**90.7**	**86.0**	**88.3**
RandomForest	Spanish	88.4	70.5	78.4
LSTM	Spanish	86.0	78.8	82.2



a) English

b) Russian

c) French

d) Spanish

Fig. 3. Heatmaps with the most frequent lexico-grammatical features of authors

anaphora. Vodolazkin uses epiphora (7.3 %) as frequently as anaphora (7.2 %).

French texts contain many diacopes: its percentage is 80–91 % that is the highest result among all the languages. Polysyndeton is significantly less popular: only 4–13 %. St Exupery differs from other authors, because he uses epiphora (5.7 %) slightly more frequently than anaphora (4.2 %).

Spanish authors have 64–77 % of diacope and 12–27 % of polysyndeton. Other features is 0.3–3 % in most cases. The exception is Valle whose texts contain 5.1 % of anaphora and 2.9 % of epiphora that significantly differs from the others.

Thus, for all languages we discover common tendencies in authors' style. Nevertheless, many authors have quite homogeneous rhythm in terms of statistics or their rhythm varies from text to text but differs from the others. Therefore, the rhythm figures seem prominent to distinguish authors.

VI. Authorship verification

For the authorship verification we apply the rhythm features and three supervised classifiers. For each author we calculate precision, recall, and F-measure. Then, to estimate verification quality for language in a whole, we calculate average precision and recall. The F-measure is computed as the harmonic mean for precision and recall.

TABLE II. Verification of English authors with AdaBoost

Classifier	Precision	Standard deviation	Recall	Standard deviation	F-measure	Standard deviation
Kingsley	**93.9**	**4.7**	**88.1**	**12.1**	**87.4**	**2.7**
Pratchett	**89.3**	**8.8**	**89.9**	**5.3**	**89.3**	**4.3**
Chesterton	87.9	10.6	80.3	3.6	81.9	7.0
McEwan	89.2	8.1	75.7	14.8	70.2	12.5
Eliot	73.3	17.1	60.3	6.4	61.7	13.0
Smith	63.8	9.5	64.9	10.5	61.5	11.3
Trollope	**89.5**	**7.0**	**84.2**	**3.9**	**86.9**	**7.1**
Atkinson	74.1	16.4	69.9	8.4	74.5	15.4
Parsons	85.1	9.3	72.1	12.8	71.6	11.4
Maugham	81.0	5.2	65.2	9.4	75.5	11.6
James	86.7	12.4	76.3	8.6	76.4	10.0
Moyes	62.0	10.8	61.6	11.1	65.8	7.4
Hardy	75.4	14.8	71.4	10.0	73.2	4.1
Henty	**89.2**	**6.4**	**81.3**	**5.7**	**86.4**	**5.0**
Rowling	77.3	12.1	59.4	2.6	73.5	7.7
Scott	**89.4**	**8.4**	**82.4**	**4.5**	**86.9**	**4.0**
Gaiman	84.8	11.0	86.0	7.6	75.3	8.1
Bindloss	**93.3**	**6.1**	**92.9**	**8.4**	**92.8**	**2.4**
Collins	76.9	17.5	69.8	6.6	73.7	7.9
Lang	77.9	8.5	72.2	4.3	72.3	14.2

TABLE III. Verification of Russian authors with AdaBoost

Classifier	Precision	Standard deviation	Recall	Standard deviation	F-measure	Standard deviation
Makanin	**87.9**	**4.5**	**88.6**	**14.3**	**87.7**	**7.7**
Gor'kij	85.1	8.3	74.4	9.4	81.8	5.3
Gogol'	68.8	24.5	53.9	5.1	62.4	12.9
Prohanov	**97.7**	**2.6**	**93.4**	**8.4**	**96.8**	**4.0**
Slavnikova	**96.7**	**3.8**	**93.1**	**4.6**	**95.0**	**6.2**
Rubanov	90.9	9.3	73.2	2.4	83.2	8.8
Vodolazkin	**96.1**	**5.0**	**88.5**	**8.1**	**86.1**	**5.7**
Rubina	86.0	8.3	69.3	7.9	73.7	8.2
Aksenov	74.3	14.3	61.6	14.1	68.8	7.8
Dostoevskij	**97.0**	**2.8**	**91.4**	**3.8**	**91.7**	**1.7**
Solzhenicyn	87.7	13.5	77.2	11.0	74.7	8.7
Tolstoy Aleksej	85.3	8.2	82.6	10.2	79.2	2.9
Nabokov	92.1	8.1	70.4	11.1	69.2	6.1
Strugackie	81.4	11.7	69.8	2.6	70.2	10.0
Pikul'	**94.1**	**6.6**	**82.9**	**8.0**	**86.8**	**10.0**
Bulgakov	68.6	19.3	65.1	12.2	63.5	13.2
Turgenev	73.7	3.9	75.6	11.5	67.9	7.0
Tolstoy Lev	76.3	12.6	65.9	8.3	71.5	14.5
Leskov	87.8	6.3	73.6	5.9	80.3	9.2
Pelevin	85.8	7.4	74.2	7.5	78.7	9.2

Table I allows to compare the classification quality for all text corpora. In every case AdaBoost outperforms the others by 10–30 % of F-measure. The precision, recall, and F-measure reach 82–90 %, 74-86 %, and 78–88 % correspondingly. The RandomForest algorithm shows the lowest result. The LSTM neural network does not achieve as high quality as AdaBoost. Most probably, it happens because of corpora sizes that are relatively small for neural networks.

The Spanish texts are verified better than the texts from other corpora: 88.3 % of F-measure. Other languages have close classification quality of F-measure 78.5–80.7 %.

Tables II, III, IV, and V show verification results for particular authors and languages. Precision, recall, and F-measure are mean values of cross-validation. Columns “Standard deviation” contain standard deviation of cross-validation results of the measure in the left column. They contain only the AdaBoost classification results that are the best among the all classifiers.

Among the English authors, Kingsley, Pratchett, Trollope, Henty, Scott, and Bindloss have the best F-measure from 86.4 % to 92.8 %. Their F-measure deviations are also quite low: 2.4–7.1 %. Texts of Eliot, Smith, and Moyes are verified with the lowest results of 61.5–65.8 % of the F-measure and 7.4–13.0 % of the standard deviation. Other authors have the F-measure 70.2–81.9 % that is relatively high.

Among the Russian authors, Makanin, Prohanov, Slavnikova, Vodolazkin, Dostoevskij, and Pikul' have the F-measure higher than 85 % and up to 96.8 %. Their texts also achieve low standard deviations except texts of Pikul'. The texts by Gogol' and Bulgakov are classified with the lowest F-measure 62.4–63.5 % with high standard deviation. The verification of other authors is performed as good and

TABLE IV. Verification of French authors with AdaBoost

Classifier	Precision	Standard deviation	Recall	Standard deviation	F-measure	Standard deviation
Rolland	74.7	8.6	66.6	7.6	64.5	9.1
Pancol	**87.2**	**10.0**	**74.5**	**6.5**	**84.1**	**3.9**
Zola	81.5	7.6	81.0	9.1	80.2	5.7
Flaubert	**97.6**	**3.8**	**91.5**	**8.0**	**93.3**	**2.5**
St Exupery	85.8	19.4	68.8	19.0	73.4	16.8
Modiano	88.0	6.0	76.1	7.6	81.6	6.3
Cusset	85.4	8.0	77.4	10.6	72.7	16.1
Hugo	83.8	13.3	63.6	6.4	69.5	3.0
Levy	82.2	6.8	80.6	7.7	79.2	8.0
Beigbeder	69.6	13.7	68.2	5.4	69.4	5.5
Balzac	69.0	12.0	53.1	3.2	62.6	9.1
Gide	88.3	9.7	71.8	4.9	78.3	9.8
Musso	75.8	11.7	73.7	12.0	74.6	10.0
Nothomb	92.7	8.0	65.5	8.5	70.8	11.7
Proust	**90.6**	**8.8**	**90.7**	**8.5**	**89.7**	**3.6**
Verne	91.5	9.9	80.5	8.2	82.1	10.5
France	85.7	6.4	74.7	3.2	82.9	5.3
Maupassant	85.1	13.5	70.4	11.2	70.6	12.1
Colette	**94.0**	**2.0**	**89.9**	**3.7**	**89.0**	**5.4**
Gard	80.8	8.9	69.9	6.1	72.5	7.2

TABLE V. Verification of Spanish authors with AdaBoost

Classifier	Precision	Standard deviation	Recall	Standard deviation	F-measure	Standard deviation
Valle	96.1	3.6	90.8	7.4	90.0	5.7
Dicenta	78.8	25.0	60.0	13.3	81.2	19.0
Ibáñez	**95.7**	**3.8**	**94.6**	**2.7**	**93.2**	**3.1**
Galdoz	87.7	4.4	84.9	4.4	86.5	6.6
Becquer	**95.9**	**4.8**	**90.0**	**10.4**	**94.2**	**3.9**
Valera	90.6	7.5	88.1	8.8	89.8	6.2
Pereda	94.0	9.9	91.1	4.1	91.8	3.5
Bazán	86.5	6.6	88.5	5.0	87.2	3.8

diverse as in English corpora.

Among the French authors, Pancol, Flaubert, Proust, and Colette have the best F-measure from 84.1 % to 93.3 %. Their F-measure deviations are very low: 2.5–5.4 %. Rolland and Balzac are verified with the lowest results of the F-measure 62.6–64.5 % and not very high standard deviation 9.1 %. The verification of St Exupery's texts became the most unstable: standard deviations are 16.8–19.4 %.

All the Spanish authors are verified very good. Their F-measure achieve 81.2–94.4 %. Texts of Ibáñez and Becquer are classified with the largest quality higher than 90 % of all measures and the very low standard deviation of 2.7–4.8 %. The verification of Dicenta's texts became the most unstable: standard deviations are 13.3–25.0 %.

These results are the highest ones among all corpora. The reason can be the smaller number of the authors that are different by rhythm statistics as we can see in Fig. 2d and 3d.

Thus, we can see the common tendencies in the verification of the authors in different languages. For English, Russian, and French corpora we find several authors who are verified with quite high quality using only rhythm features. So their rhythm is a sign of their uniqueness, and they use the similar rhythm for many texts.

Besides, in each corpora we can see the authors who are verified with low measures 60–69 % or with the high standard deviation 10–20 %. It means that the chosen statistical features do not represent the style of these authors as unique, and their texts significantly vary in style in terms of lexico-grammatical figures.

VII. Discussion

The most significant result of our experiments with the authorship verification is that the use of only rhythm features of a literary text in vectors for the classification leads to the high classification results, the average F-measure is above 78 %. It shows that the features of the prose rhythm are useful markers of the author's style, along with such popular stylometric features as n-grams, morphological features, and syntactic features.

Another important result of our study is that rhythm features can be easily interpreted from the point of view of a linguist. Automatic search of lexico-grammatical figures and, as a consequence, statistical analysis of the large number of texts, as well as additional visualization of these results, brings expert linguistic analysis to a new level. A linguist is able to simultaneously evaluate the whole range of features for many works and authors. In our study, we assessed the rhythm both for individual authors and for the language as a whole.

The rhythm features are most actively used in English and Spanish. This is well seen from the frequencies of diacope and other rhythmic features. French features were in the minority, and their values do not exceed the frequency of features in other languages.

The diacope is the most frequently used rhythm figure. In the Russian language, we observe the smallest percentage of diacope among all rhythm figures, so we can conclude that the rhythm is essentially achieved by other features. Interestingly, the percentage of diacope differs in different languages. In Russian texts — 56.0 ± 3.9 %, in English texts — 65.9 ± 3.1 %, in Spanish texts — 70.8 ± 2.9 %, and in French texts - 85.5 ± 2.5 %. We can hypothesize that each language, existing within the framework of its own linguistic rules, produces a stable ratio of diacope and the total number of all features, regardless of the author. This may indicate that the rhythm indicators, in particular the diacope and its use, contribute to the recognition of the rhythm structure of the language as a whole.

The expert linguistic analysis of the authorship verification results is an additional large task and the subject of the next study. In particular, it is the analysis of the authorship verification errors.

VIII. Conclusion

The quality of result of the authorship verification based on the rhythm features of the text is quite high and achieves up to 86–97 % of all measures for many authors. It shows that the rhythm features are significant markers of the author's style of a text. An additional advantage of these parameters is the possibility of an expert evaluation of the results of numerical experiments by linguists.

The obtained data open up broad perspectives for research in both computer and classical linguistics. The calculation of the frequency of rhythm features allows working with the problems of the authorship of the text, determining the specifics of the author's style. The statistical analysis and visualization of rhythm makes it possible for linguists to solve a number of large-scale problems in the field of determining the dynamics of the literary process as a whole, peculiarities of rhythm indicators in different languages.

Acknowledgment

The reported study was funded by RFBR according to the research project No. 19-07-00243.

References

[1] J. S. Li, L.-C. Chen, J. V. Monaco, P. Singh, and C. C. Tappert, "A comparison of classifiers and features for authorship authentication of social networking messages," *Concurrency and Computation: Practice and Experience*, vol. 29, no. 14, p. e3918, 2016.

[2] A. Altamimi, N. Clarke, S. Furnell, and F. Li, "Multi-platform authorship verification," in *Proceedings of the Third Central European Cybersecurity Conference*, 2019, pp. 1–7.

[3] O. Halvani and L. Graner, "Rethinking the evaluation methodology of authorship verification methods," in *International Conference of the Cross-Language Evaluation Forum for European Languages*. Springer, 2018, pp. 40–51.

[4] O. Halvani, L. Graner, and R. Regev, "Taveer: an interpretable topic-agnostic authorship verification method," in *Proceedings of the 15th International Conference on Availability, Reliability and Security*, 2020, pp. 1–10.

[5] J. A. Stover, Y. Winter, M. Koppel, and M. Kestemont, "Computational authorship verification method attributes a new work to a major 2nd century a frican author," *Journal of the Association for Information Science and Technology*, vol. 67, no. 1, pp. 239–242, 2016.

[6] E. Tuccinardi, "An application of a profile-based method for authorship verification: Investigating the authenticity of pliny the younger's letter to trajan concerning the christians," *Digital Scholarship in the Humanities*, vol. 32, no. 2, pp. 435–447, 2017.

[7] S. Corbara, A. Moreo, F. Sebastiani, and M. Tavoni, "The epistle to cangrande through the lens of computational authorship verification," in *International Conference on Image Analysis and Processing*. Springer, 2019, pp. 148–158.

[8] M. Kestemont, G. Martens, and T. Ries, "A computational approach to authorship verification of Johann Wolfgang Goethe's Contributions to the Frankfurter gelehrte Anzeigen," *Journal of European Periodical Studies*, vol. 4, no. 1, pp. 115–143, 2019.

[9] M. A. Al-Khatib and J. K. Al-qaoud, "Authorship verification of opinion articles in online newspapers using the idiolect of author: a comparative study," *Information, Communication & Society*, pp. 1–19, 2020.

[10] K. Lagutina, N. Lagutina, E. Boychuk, I. Vorontsova, E. Shliakhtina, O. Belyaeva, I. Paramonov, and P. Demidov, "A survey on stylometric text features," in *Proceedings of the 25th Conference of Open Innovations Association (FRUCT)*. IEEE, 2019, pp. 184–195.

[11] H. Gómez-Adorno, G. Sidorov, D. Pinto, D. Vilariño, and A. Gelbukh, "Automatic authorship detection using textual patterns extracted from integrated syntactic graphs," *Sensors*, vol. 16, no. 9, p. 1374, 2016.

[12] K. Lagutina, N. Lagutina, E. Boychuk, and I. Paramonov, "The influence of different stylometric features on the classification of prose by centuries," in *Proceedings of the 27th Conference of Open Innovations Association FRUCT*. IEEE, 2020, pp. 108–115.

[13] E. Boychuk, I. Paramonov, N. Kozhemyakin, and N. Kasatkina, "Automated approach for rhythm analysis of French literary texts," in *Proceedings of 15th Conference of Open Innovations Association FRUCT*. IEEE, 2014, pp. 15–23.

[14] M. Kestemont, E. Manjavacas, I. Markov, J. Bevendorff, M. Wiegmann, E. Stamatatos, M. Potthast, and B. Stein, "Overview of the cross-domain authorship verification task at pan 2020," in *CLEF*, 2020.

[15] M. L. Brocardo, I. Traore, I. Woungang, and M. S. Obaidat, "Authorship verification using deep belief network systems," *International Journal of Communication Systems*, vol. 30, no. 12, p. e3259, 2017.

[16] O. Halvani, C. Winter, and A. Pflug, "Authorship verification for different languages, genres and topics," *Digital Investigation*, vol. 16, pp. S33–S43, 2016.

[17] H. Ahmed, "The role of linguistic feature categories in authorship verification," *Procedia computer science*, vol. 142, pp. 214–221, 2018.

[18] S. Adamovic, V. Miskovic, M. Milosavljevic, M. Sarac, and M. Veinovic, "Automated language-independent authorship verification (for indo-european languages)," *Journal of the Association for Information Science and Technology*, vol. 70, no. 8, pp. 858–871, 2019.

[19] B. Boenninghoff, S. Hessler, D. Kolossa, and R. M. Nickel, "Explainable authorship verification in social media via attention-based similarity learning," in *2019 IEEE International Conference on Big Data*. IEEE, 2019, pp. 36–45.

[20] M. King and P. Cook, "Authorship verification with personalized language models," in *International Conference on Text, Speech, and Dialogue*. Springer, 2020, pp. 248–256.

[21] B. Boenninghoff, R. M. Nickel, S. Zeiler, and D. Kolossa, "Similarity learning for authorship verification in social media," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 2457–2461.

[22] N. E. Benzebouchi, N. Azizi, M. Aldwairi, and N. Farah, "Multi-classifier system for authorship verification task using word embeddings," in *2018 2nd International Conference on Natural Language and Speech Processing (ICNLSP)*. IEEE, 2018, pp. 1–6.

[23] K. Lagutina, A. Poletaev, N. Lagutina, E. Boychuk, and I. Paramonov, "Automatic extraction of rhythm figures and analysis of their dynamics in prose of 19th-21st centuries," in *Proceedings of the 26th Conference of Open Innovations Association FRUCT*. IEEE, 2020, pp. 247–255.

[24] E. Boychuk, I. Vorontsova, E. Shliakhtina, K. Lagutina, and O. Belyaeva, "Automated approach to rhythm figures search in English text," in *International Conference on Analysis of Images, Social Networks and Texts, Communications in Computer and Information Science*, vol. 1086. Springer, 2019, pp. 107–119.

[25] K. Kowsari, K. Jafari Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown, “Text classification algorithms: A survey,” *Information*, vol. 10, no. 4, pp. 150 (1–68), 2019.

[26] M. Sokolova and G. Lapalme, “A systematic analysis of performance measures for classification tasks,” *Information processing & management*, vol. 45, no. 4, pp. 427–437, 2009.