






Authorship Attribution using Sequential Part-of-Speech Pattern Mining

Sirisha Alamanda^{1*}, Suresh Pabboju², Narasimha Gugulothu¹

¹ Department of Computer Science, Jawaharlal Nehru Technological University (JNTU-H), Hyderabad 500085, India

² Department of Information Technology, Chaitanya Bharti Institute of Technology, Hyderabad 500075, India

Corresponding Author Email: asirishanaidu38@gmail.com

<https://doi.org/10.18280/ria.370117>

Received: 18 December 2022

Accepted: 10 February 2023

Keywords:

authorship attribution, part-of-speech, n-grams, skip-grams, Part-of-Speech-Skip-Gram, sequential pattern mining, author identification

ABSTRACT

Given an anonymous text, automatically attributing a name from a group of known writers is called "Authorship Attribution" (AA). It is a classification problem, and feature extraction techniques are initially applied, followed by the training of a model using a collection of texts whose authors are known. Numerous features, such as lexical, semantic, structural, n-grams, etc., can be used to identify the stylistic characteristics of writers. The authors of this research propose a novel approach to this problem by using sequential pattern mining on part-of-speech (PoS) tags. This paper introduces and discusses the concept of a Part-of-Speech Skip-Gram (PoSSG) that is different from traditional n-gram. A sequential pattern mining algorithm is applied to obtain PoSSG patterns, which are then used for authorship attribution tasks. Experimental studies on two different datasets: novels extracted from Project Gutenberg and Stamatatos06 Author Identification: C10-Attribution confirms that this approach of mining PoSSG patterns facilitates author identification.

1. INTRODUCTION

Authorship Attribution (AA) is a classification challenge in which the author of an anonymous text (a writing without a label) must be determined among a group of potential writers [1]. An example used frequently in AA is the need to identify the author of each newly discovered historical text. We could also find out more about the author's country of origin or the genre of the text [2].

For AA, numerous methods have been devised. But selecting suitable attributes to accurately label a group of texts is a major challenge for AA. There are a lot of suggested attributes, but none of them is well known to be completely correct in every circumstance [3]. Because of this, it is essential to come up with new indicators and ways to improve authorship attribution systems in the modern world.

In order to effectively define an author's style of writing, this study examines a novel stylistic element known as a Part-of-Speech Skip-Gram (PoSSG) and takes it into consideration for authorship attribution [4]. Like a part-of-speech (PoS) n-gram, a PoSSG can be built with a skip distance or gap between consecutive PoS tags to show how often an author uses a certain sentence form [5].

In contrast to previous research that used definite-size PoS n-grams for author identification, the proposed approach uses skip-grams, which allow gaps between PoS tags of sentences to match a person's personal style well [6]. Additionally, the proposed approach, in contrast to some earlier studies, simply retrieves the most prevalent k skip-grams instead of determining the frequencies of all POS skip-grams, speeding up processing [7]. Finally, by finding PoSSGs of varied lengths and gaps, the proposed approach is more adaptable.

Two different datasets, one with a set of 30 texts and another with 500 texts by 10 different authors, are used to describe the experimental studies. The study supports the idea that

identifying PoSSG patterns in texts makes determining authorship easier [8]. Additionally, studies demonstrate that employing gaps yields significantly better results as compared to PoS bigrams and trigrams. Finally, a noteworthy finding is that good classification accuracy can be achieved by employing a small collection of the 50–100 most common skip-grams in various sizes [9].

The remaining portions of this study are structured as follows: Related research is discussed in Section 2. The proposed approach and the datasets used are explained in Section 3. Section 4 describes the research findings. Finally, Section 5 presents conclusions.

2. RELATED WORKS

Law enforcement may be able to save lives by identifying the writer of a text (such as a suicide note or a phone message); thus, AA continues to be a significant challenge in the fields of information retrieval and journalism. Stamatatos et al. [1] came up with a statistical method based on a vector of 22 stylistic indicators that can be used to figure out who wrote a newspaper article using statistics.

Three machine learning methods were examined by Zheng et al. [2] using a variety of features, including style markers (SM), structural features (SF), and content-specific features (CF). To identify an author's style, finding the right combination of features has been the subject of several studies [3]. Their analysis yielded predictions with an average accuracy of 80%–90% for email messages and 90%–97% for news messages. They discovered, however, that feature selection is important for attaining maximum scores, and compared to other models, SVM performed better. Houvardas and Stamatatos [4] have achieved an accuracy of 100% for the small English corpus by using SVM with word frequencies

and n-grams on several corpora. More parameters, lexical features, and 1,000 syntactic features were added by Van Halteren [5].

According to Stamatatos [6], sentence splitters, tokenizers, PoS taggers, and other tools are widely used in authorship attribution models. To identify the author of some emails, Koppel et al. [7] integrated lexical (functional words), collocational (bi-gram sections of speech), and individualistic data. A variety of syntactic n-grams (sn-grams) have been employed by Sidorov et al. [8]; these include PoS n-grams, character n-grams, and sn-grams, where the elements are chosen based on their position in the syntactic tree rather than their order of presentation in the text. Three authors' 39-document corpus served as the basis for their research, which demonstrated that sn-grams are more successful than conventional n-grams because they place greater emphasis on the relationships among words' syntactic structures. The significance of content pre-processing in AA has been studied by Markov et al. [9] using a language-dependent tool called Entity Recognition (NER).

A document with the greatest score (similarity) is produced by Mitra and Craswell [10] by using the similarity function to determine how similar a query article is to each other article in the corpus. A writing style analysis-based method for the purpose of identifying the authors of papers with a lot of identical content has been proposed by Rexha et al. [11]. In the study [12], Zhang et al. also showed a new way to figure out who wrote short texts by using an author-document topic model.

Bacciu et al. [13] employed character, word n-grams, stemmed words, and distorted text in a cross-domain setting covering four different languages: French, Italian, English,

and Spanish. An SVM model was used for each feature, and an ensemble architecture achieved a F1-score of 68%. Pretrained language models have been studied by Barlas and Stamatatos [14] and Fabien et al. [15] for their potential use in identifying the author of texts. The basic process by which trained language models like [16] generate contextualised word embeddings involves providing a sequence of words as input. One can determine the context and identify the writer of each text using these embeddings. Even with simpler embedders, embedding-based methods could be equally efficient [17].

Pizarro [18] introduced a SVM model with character and word n-grammes to assess whether the author of a Twitter feed is eager to disseminate fake news and achieved an average accuracy of 0.7775–0.7350 for English. When using the AA method as a statistical approach, Khomytska and Teslyuk [19] used three hypothesis testing techniques: the chi-square test, the Kolmogorov-Smirnov test, and the student's test. Contrarily, Custódio and Paraboni [20] have created a stacked model by using a few classifiers to produce a model that is reliable in a variety of domains, languages, and contexts.

Table 1 shows the importance of n-grams in stylometric tasks, and many of the previous works have used SVM to improve accuracy on textual problems. The previous studies have been conducted on different varieties of corpora like blogs, emails, messages, and books using machine learning approaches, whereas this study mainly focuses on large books and articles using a sequential pattern mining approach [21]. The proposed work is an effort to probe further into sentence structure while using the PoS tags to distinguish across authors' writing styles to analyse the impact of this feature on the final assessment [22].

Table 1. Summary of related works

Ref	Corpus	Feature	Algorithm	Accuracy%
[4]	Small size English corpus	n-gram	SVM	100
		bigrams	SVM	85-95
[8]	39 documents by three authors	trigram	SVM	71-95
		sn-grams	SVM	100
			n-gram text model	100
[17]	PAN12dataset with a test set of 6 documents	n-gram	n-gram TAG model	83
			Functional words model	66.6
	Gutenberg books			85-70
	Poetry			60-40
[21]	IMDB	TF-IDF	TF-IDF based	80-50
	Blogs		AARR_W model	90-75
	PAN2011			60-25
	Twitter			80-50

The unique method described in this study is carried out in three basic steps. First, part-of-speech taggers [23] are used to pre-process a collection of training texts from well-known authors. After that, each training text is mined using a top-k sequential pattern mining method developed by Fournier-Viger et al. [24] to uncover the most common k PoSSG patterns. The distinctive signature that represents the writing style of an author is then developed using these PoSSG patterns. Finally, anonymous texts are classified using the retrieved signatures.

3. METHODOLOGY

Finding the most probable writer of a previously undiscovered text with unidentified authorship utilising a sample of reference texts from a limited group of potential writers is the goal of the AA challenge. In the proposed work, a training corpus of labelled texts represented as C by m authors was considered as input. Let the candidate authors be denoted as $A = \{A_1, A_2, \dots, A_m\}$. Assume that every author A_i , where $1 \leq i \leq m$, wrote a collection of n writings denoted by $T_i = \{t_1, t_2, \dots, t_n\}$, for a total of $m \cdot n$ texts in the corpus. The four phases of the proposed work to find the author of an unknown text are discussed in the following subsections:

3.1 Extracting PoS tags from a text

In the first stage, illustrations, punctuation, and other non-authorial-style content are removed from each text document inside the corpus through pre-processing. Then, the Rita Natural Language Processing library [22] or the Stanford NLP tagger [23] are used to replace each word of a sentence with one of 36 PoStags because the proposed strategy is based on PoS tags.

Consider the following paragraph: “The band major was a poet. His name is lost to history, but it deserves a place among the titles of the great. Only in the soul of a poet, a great man, could there have been conceived that thought by which the music of triumph should pass the little pinnacle of human exultation, and reach the higher plane of human sympathy.” from the novel *The Girl at the Halfway House* by Emerson Hough.

After extracting the PoS tags from the above paragraph, the sentences of the paragraph seem like the following PoS sequences: DT NN JJ VBD DT.

PRP\$ NN VBZ VBD TO NN CC PRP VBZ DT NN IN DT NNS IN DT.

RB IN DT NN IN DT NN DT JJ NN MD EX VBP VBN VBN IN VBD IN WDT DT NN IN NN MD VB DT JJ NN IN JJ NN CC VB DT JJR NN IN JJ.

3.2 Extracting author’s unique sequential PoS patterns

This stage of the proposed work has two tasks. First, mining top-k frequent sequential PoSSG patterns from every corpus text t of an author. Second, combining the frequent sequential PoSSG patterns of all the texts in the corpus that belong to author A_i to form a unique set of sequential PoS patterns for the author.

The sequential PoSSG patterns are the same as part-of-speech n-grams, which allow a gap between contiguous elements. To extract the frequent PoSSG patterns, the

proposed approach considered four parameters: the sequential PoSSG pattern’s minimum length nl , the maximum length xl , number of PoSSG patterns to be found k , and the maximum gap allowed among contiguous PoS tags in the PoSSG pattern, max_gap .

Definition 1. PoSSG: Consider a sentence with the PoStags p_1, p_2, \dots, p_p and the parameter max_gap (a positive integer). An ordered list of n tags $p_{i_1}, p_{i_2}, \dots, p_{i_n}$ where $i_1, i_2, i_3, \dots, i_n$ are integers such that $i_j - (i_{j-1}) \leq max_gap + 1$ and $1 < j \leq n$ is known as a n -skip-gram. One should be aware that PoS n-grams are a specific instance of PoSSG when $max_gap = 0$. (i.e., no gaps).

3.2.1 Mining the frequent part-of-speech skip-gram (PoSSG) patterns

For every text t in the training corpus, the top-k sequential PoS skip-gram patterns are determined using the approach proposed by Fournier-Viger [24]. Here each PoS tag represents an item, and each sentence represents a sequence. The percentage of the total number of sentences in the text that contain a PoS-skip-gram (PoSSG) is used to calculate a PoSSG frequency. Figure 1 depicts the precise steps of his phase.

In this study, the top-k frequently occurring PoS skip-gram patterns of length between nl and xl in a corpus text t are called part-of-speech skip-gram patterns, abbreviated as $(PoSSG_{A_i})_{nl,xl}^k$.

Consider the example paragraph presented in the previous phase. The set $(PoSSG_{A_i})_{1,2}^5$ which represents the top-5 frequent PoSSG patterns of length between $nl = 1$ and $xl = 3$ for a $max_gap = 1$ of the text are shown in Table 2.

Table 2. PoSSGs discovered in the sample paragraph

PoSSG found	Description	Frequency%
NN-IN	Noun, singular or mass -Preposition or subordinating conjunction	66.7
IN	Preposition or subordinating conjunction	66.7
DT-JJ	Determiner - Adjective	66.7
NN-CC	Noun, singular or mass	66.7
IN-DT	Preposition or subordinating conjunction -Determiner	66.7
CC	Coordinating conjunction	66.7
JJ	Adjective	100
VBD	Verb, past tense	100
DT	Determiner	100
NN	Noun, singular or mass	100
DT-NN	Determiner - Noun, singular or mass	100

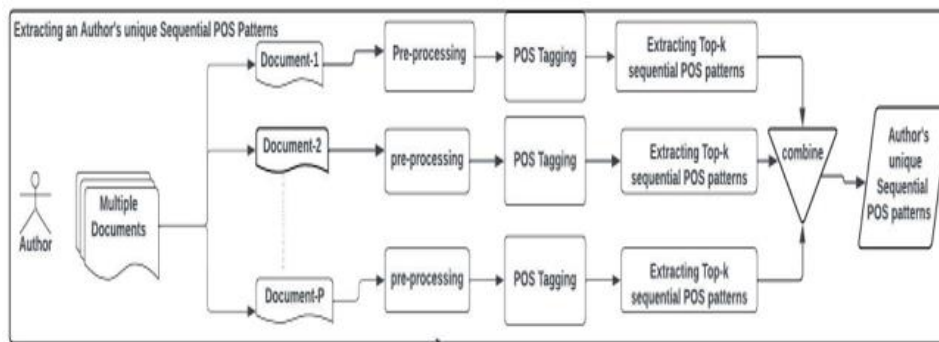


Figure 1. Extracting unique PoSSG patterns of an author

Permitting a max_gap of 1 tag in a skip-gram between two adjoining tags in the above example enables the identification of DT-JJ, which is available in two sentences. Therefore, the frequency of this skip-gram is 66.7%. It is significant to observe that the pattern DT-JJ is not found as frequent standard PoS n-grams like PoS bigrams. Here are the instances of the skip-gram DT-JJ emphasized as: NN **JJ** VBD DT.

PRP\$ NN VBZ VBD TO NN CC PRP VBZ DT NN IN DT NNS IN DT.

RB IN DT NN IN DT NN **DT JJ** NN MD EX VBP VBN VBN IN VBD IN WDT DT NN IN NN MD VB DT JJ NN IN JJ NN CC VB DT JJR NN IN JJ.

3.2.2 Combining PoSSG patterns of all the texts of an author

During this task, we compute the unique PoSSG patterns of an author. For each author A_i the top-k PoSSG patterns of all the texts in the corpus of an author extracted through the previous task are combined.

Definition 2: The unique set of PoSSSG patterns of an author A_i abbreviated as $(PoSSG_{A_i})_{nl,xl}^k$ is computed by Eq.

(1), to be the union of frequent PoSSG patterns obtained from the set of writings $T_i = \{t_1, t_2, \dots, t_n\}$ of the author in the corpus.

$$(PoSSG_{A_i})_{nl,xl}^k = \bigcup_{t \in T_i} (PoSSG_t)_{nl,xl}^k \quad (1)$$

Two very different types of patterns may be found with exceptional frequency in the PoSSG patterns of the author A_i : one that represents conventional English sentence structures and the other that really identifies the author's writing style. By finding the intersection of all of the authors' PoSSG patterns, a common set of PoSSG patterns is found that can be used to make each author's signature for authorship attribution.

Definition 3: The common set of part-of-speech skip-gram patterns of all authors, abbreviated as $(CPoSSG)_{nl,xl}^k$ is the intersection of PoSSG patterns of all the m authors in the corpus is computed by Eq. (2).

$$(CPoSSG)_{nl,xl}^k = \bigcap_{i=1}^m (PoSSG_{A_i})_{nl,xl}^k \quad (2)$$

3.3 Creating an author's unique signature

After finding the CPoSSG of all authors, now the unique signatures of all the authors, which truly characterize their unique style, are computed by excluding the common PoS skip-gram patterns from each author's unique set of PoSSG patterns extracted in the previous step. Figure 2 explains the detailed process of this phase. This step is repeated for all the authors to extract their unique signatures.

Definition 4: After extracting the CPoSSG patterns of all authors, the unique signature of an author A_i denoted as S_{A_i} is computed by Eq. (3).

$$(S_{A_i})_{nl,xl}^k = (PoSSG_{A_i})_{nl,xl}^k - (CPoSSG)_{nl,xl}^k \quad (3)$$

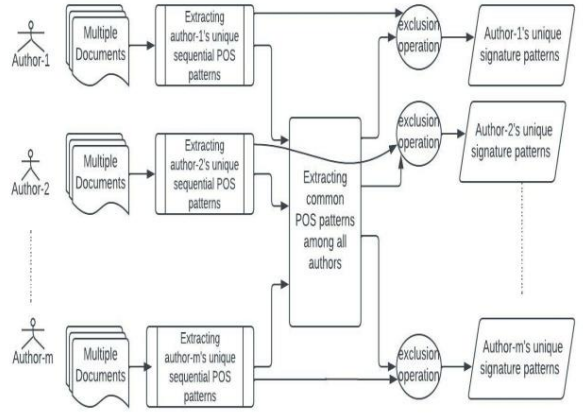


Figure 2. Creating unique signatures for authors

3.4 Finding the author of an unknown text

The retrieved authors signatures from the previous phase are used to classify unknown texts in the last phase of the proposed work. In order to determine the author a_u of an unknown text t_u which is not utilised during training, a classification algorithm AAPoSSG-patterns are designed. The set $S^1 = \{S_{A_1}, S_{A_2}, S_{A_3}, \dots, S_{A_m}\}$ of author signatures, an anonymous text (t_u), and the parameters (nl, xl, max_gap , and k) are all input to the AAPoSSG-patterns algorithm. In the anonymous text t_u , first the algorithm extracts the top-k PoSSG patterns. Then, using a similarity function, it contrasts the patterns in t_u with the signatures of each author. In a sorted map, for every author, a record is listed with the author's name along with the value of his similarity to the anonymous text. At last, the algorithm outputs this map, which is ordered by decreasing similarity. The anonymous text t_u 's most likely authors are ranked on this map. Figure 3 gives an overview of this phase.

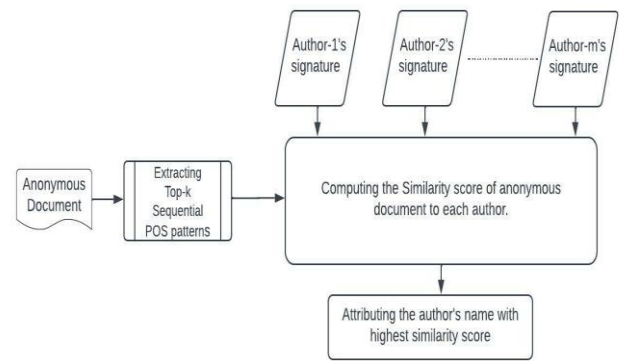


Figure 3. Authorship attribution process for an unknown text

Different metrics, including Euclidian distance, cosine similarity, the Jaccard coefficient, and Sorenson-Dice, can be employed to measure the similarity. As the proposed study is based on sets of PoSSG patterns, the Jaccard coefficient (JC) and Sorenson-Dice, (SD) are chosen to measure the similarity score between the PoSSG sets. And from the experimental results JC was chosen as the optimal metric.

Algorithm: AAPoSSG-patterns
<p><i>Input:</i> The set of signatures of all the authors $S^1 = \{S_{A_1}, S_{A_2}, S_{A_3}, \dots, S_{A_m}\}$, unknown author's text t_u, the parameters nl, xl, max_gap, no. of required patterns k.</p> <p><i>Output:</i> A sorted map M of the most likely authors of t_u along with their scores.</p>
<p>Step 1: Extract the top-k PoSSG patterns of the unknown text t_u, as per the parameters nl, xl, max_gap, k i.e $(PoSSG_{t_u})_{nl,xl}^k$ using the top-k sequential pattern algorithm.</p> <p>Step 2: create a Sorted Map M to store (similarity score and author name) as key, value pairs.</p> <p>Step 3: Compute the similarity score of the unknown text to each author's signature.</p> <p>For each $(S_{A_i})_{nl,xl}^k$ in the signatures set S^1 do</p> <p>Compute the similarity score JC_{A_i} between the two sets $(PoSSG_{t_u})_{nl,xl}^k$ and $(S_{A_i})_{nl,xl}^k$ using the Jaccard coefficient.</p> <p>Insert (JC_{A_i}, A_i) to M.</p> <p>Step 4: return M sorted by the decreasing similarity score. End.</p>

The Jaccard coefficient (or index) measures the similarity between two sets. In this work, the Jaccard similarity coefficient JC_{A_i} between an unknown text t_u and an author's signature S_{A_i} is computed by Eq. (4). The vertical bar on either side refers to the size of a set.

$$JC_{A_i} = \frac{|(PoSSG_{t_u})_{nl,xl}^k \cap (S_{A_i})|}{|(PoSSG_{t_u})_{nl,xl}^k \cup (S_{A_i})|} \quad (4)$$

Similarity The Sorenson-Dice index denoted as SD_{A_i} is another tool for determining the similarity of two sets. In this work, the Sorenson-Dice similarity SD_{A_i} between an unknown text t_u and an author's signature S_{A_i} is computed by Eq. (5).

$$SD_{A_i} = 2 * \frac{|(PoSSG_{t_u})_{nl,xl}^k \cap (S_{A_i})|}{|(PoSSG_{t_u})_{nl,xl}^k| + |(S_{A_i})|} \quad (5)$$

Another criterion, the success ratio, abbreviated as R_z , is used to assess the performance of the algorithm. R_z is the percentage of texts in the test corpus for which the actual author was determined to be one among the z authors who is most probable. R_z is computed using Eq. (6).

$$R_z = \frac{\text{Number of matches found}}{\text{Number of texts in the test corpus}} \quad (6)$$

4. EXPERIMENTAL STUDIES

To see how well the proposed method, which is based on sequential PoS skip-gram patterns, worked for authorship attribution, a series of experiments were carried out using the following two datasets.

4.1 Canadian authors novel dataset

This is a collection of 30 novels from the 19th century that were authored by 10 different English novelists. The books are downloaded from a website called Project Gutenberg [25]. It is a repository of public domain books that are given as text files. The books weren't chosen at random. Instead, they were chosen because they were written contemporaneously by authors from the same country around the same time. Authors who wrote at least three novels for Project Gutenberg were selected. Table 3 shows the total words and sentences in each author's novel collection.

Each text from the corpus was pre-processed to remove non-authorial-style content. Then, to learn about and evaluate how well the proposed approach worked, for each author, two texts were used to train, and one text was used to test. So, the proposed system was trained with 20 different texts from all 10 authors. The signatures of the 10 authors were determined by taking the most common PoS skip-gram patterns from all these texts. The validation was done by comparing the PoSSG patterns of each text in the test corpus and ranking the 10 author signatures from most probable to least probable. Each one of the texts in the test corpus went through this whole process.

Table 3. Canadian authors novel dataset statistics

S.No	Author name	words	sentences
1	Catharine Traill	276,829	6,588
2	Emerson Hough	295,166	15,643
3	Henry Addams	447,337	14,356
4	Herman Melville	208,662	8,203
5	Jacob Abbott	179,874	5,804
6	Louisa May Alcott	220,775	7,769
7	Lydia Maria Child	369,222	15,159
8	Margaret Fuller	347,303	11,254
9	Stephen Crane	214,368	12,177
10	Thornton W. Burgess	55,916	2,950

Table 4. The classification results on the Canadian Novels dataset using the PoSSG patterns for different k values with max_gap=1

a) k=50									
Success ratio in%									
Jaccard Coefficient (JC)					Sorensen–Dice (SD)				
nl,xl	1,2	1,3	1,4	1,5	nl,xl	1,2	1,3	1,4	1,5
R ₁	60	60	40	40	R ₁	60	50	40	40
R ₂	70	60	70	70	R ₂	70	60	70	70
R ₃	80	100	90	90	R ₃	80	70	80	80
R ₄	80	100	90	90	R ₄	80	90	90	90
R ₅	80	100	90	90	R ₅	80	100	90	90

b) k=100									
Success ratio in%									
Jaccard Coefficient (JC)					Sorensen–Dice (SD)				
nl,xl	1,2	1,3	1,4	1,5	nl,xl	1,2	1,3	1,4	1,5
R ₁	50	60	60	70	R ₁	30	40	50	40
R ₂	90	80	90	90	R ₂	80	70	80	80
R ₃	90	80	100	100	R ₃	100	90	90	90
R ₄	100	100	100	100	R ₄	100	90	100	100
R ₅	100	100	100	100	R ₅	100	100	100	100

c) k=200									
Success ratio in%									
Jaccard Coefficient (JC)					Sorensen–Dice (SD)				
nl, xl	1,2	1,3	1,4	1,5	nl, xl	1,2	1,3	1,4	1,5
R ₁	60	80	80	80	R ₁	60	80	80	80
R ₂	70	80	100	90	R ₂	70	80	100	90
R ₃	90	100	100	100	R ₃	90	100	100	100
R ₄	100	100	100	100	R ₄	100	100	100	100
R ₅	100	100	100	100	R ₅	100	100	100	100

Several experiments were done to figure out how the parameters used in this proposed method affect the success rate. These parameters include the minimum and maximum length of PoSSG patterns (nl and xl), the frequent PoSSG patterns to be found (k) from a text, and the maximum gap allowed between consecutive PoS tags in a sequential pattern (max_gap). For this experimental study, the parameters were set as max_gap with values 0, 1, and 2, and k with values 50, 100, and 200. The results for k = 200 and max_gap = 2 are not shown because there isn't enough space. The maximum length of PoSSG is varied from 2 to 5 for each value of k. Table 4 shows the results obtained for max_gap = 1, and Table 5 show the results obtained for bi-grams and trigrams when nl, xl, and k were given different values. Also in each subtable, the success ratio R_z obtained with respect to the similarity measures Jaccard Coefficient (JC) and Sorensen–Dice (SD) for different z values ranging from 1 to 5 is also presented.

From the results of Table 4, it can be observed that for max_gap = 1 and k = 50, the best results were achieved for PoSSG patterns of length between nl = 1 and xl = 3, and the similarity function Jaccard Coefficient has given better results than

Sorensen–Dice. With these parameters, 60% of anonymous test corpus texts had their true authors identified as the top-1 most likely authors (rank R₁) in the sorted list returned by the proposed algorithm, and 100% of the texts had their true authors identified as one of the top three most likely authors (R₃). Similarly, for max_gap = 1 and k = 100, the best results were achieved with the Jaccard coefficient for nl = 1 and xl = 5. With these parameters, 70% of novels are correctly attributed to the top-most probable author (R₁), 90% of texts are truly attributed to the top-two authors (R₂), and 100% to one of the top-three most likely authors (R₃). Also, when max_gap = 1 and k = 200, the Jaccard coefficient for nl = 1 and xl = 5 gave the best results, which are almost the same as the results for k = 100.

From the results, it is clear that for longer skip-grams, increasing the value of k beyond 200 does not usually lead to better results. This means that a small set of PoSSG patterns, with k ranging between 50 and 200, can represent an author's unique signature and give a good picture of the author's writing style. This is different from earlier research like that of Sidorov et al. [8], who have used 400 to 11,000 n-grams or sn-grams.

Table 5. The classification results for the Canadian Novels dataset using the bigrams and trigrams

Success ratio in% with Jaccard Coefficient (JC)						
nl, xl	K=50		K=100		K=200	
	2,2 (bigram)	3,3(trigram)	2,2 (bigram)	3,3(trigram)	2,2 (bigram)	3,3(trigram)
R ₁	70	50	70	80	60	50
R ₂	80	90	80	90	70	100
R ₃	80	100	90	90	70	100
R ₄	90	100	100	100	100	100
R ₅	100	100	100	100	100	100

Table 6. The classification results for the C10 dataset using the PoS skip-grams for different k values with max_gap = 1

Success ratio in% with Jaccard Coefficient (JC)												
nl, xl	K=50				K=100				k=200			
	1,2	1,3	1,4	1,5	1,2	1,3	1,4	1,5	1,2	1,3	1,4	1,5
R ₁	17	18	20	17.6	16	24	23.19	20	19.2	38	17.6	27.4
R ₂	29.4	32.2	34.6	33.2	30.4	36.6	37.6	36.6	30.4	55	40.2	45.4
R ₃	42.2	44	46.39	45.4	45.4	49.2	50	48.8	46.2	69	57.6	57.2
R ₄	53.6	56.8	57	57.2	58	60.39	60.8	55.6	59.8	78.4	69	66
R ₅	63.2	65.8	67	66	66.6	71	71.2	64.6	69.6	85.2	77.4	74.2

Table7. The classification results for the C10 dataset using the Bigrams and Trigrams

success ratio in% with Jaccard Coefficient (JD)						
nl, xl	K=50		K=100		K=200	
	2,2 (Bigram)	3,3 (Trigram)	2,2 (Bigram)	3,3 (Trigram)	2,2 (Bigram)	3,3 (Trigram)
R ₁	14.8	19	15.2	38.8	18.4	43.39
R ₂	30.8	37.2	30.59	54.8	30.8	60.6
R ₃	44.4	49.6	44.4	68.2	45.6	70.8
R ₄	52.2	59.2	54.4	77	57	80.6
R ₅	63.6	66.2	65.6	83.4	67.6	87.4

This work has also compared the results obtained with PoS bigrams and trigrams used in previous work [7] by Koppel and Schler. In Table 5, the results for bigrams and trigrams are shown. The best results are found with k = 100, which is pretty close to the results found with skip-grams. But the skip-grams results are better because the test corpus true authors were correctly predicted with a success ratio of 70% in R1, 90% in R2, and 100% in R3, compared to 80%, 90%, and 90% when using part-of-speech trigrams for k=100.

4.2 C10-Attribution dataset

This dataset in the study [26] contains 500 texts from 10 different authors/candidates represented as C10 each have written 50 texts. The proposed System has been trained on the 500 texts and then it has been tested using a test corpus of 500 files found inside the ground truth json-file. The dataset can be found at the link <https://doi.org/10.5281/zenodo.3759064>.

The Table 6 represents classification results for the C10 dataset using the PoS skip-grams for different k values with max_gap = 1. The results of the proposed work on this dataset clearly show that for max_gap = 1, k = 50, nl = 1, and xl = 4, the best results were obtained using the Jaccard coefficient. In this dataset, as the test corpus has the same number of texts as the training corpus, i.e., 500 texts, the success ratio for these parameters is 20% for rank R1, 34.6% for R2, 46.39% for R3, 57% for R4, and 67% for R5. Similarly, for max_gap = 1 and k = 100, the best results were achieved with the Jaccard coefficient for nl = 1 and xl = 4, which are slightly better than the results obtained for nl = 1 and xl = 3. On the other hand, for max_gap = 1 and k = 200, the best results were achieved for nl = 1 and xl = 3. With these parameters, from the output returned by the algorithm, i.e., the ranked list of most likely authors, it is observed that 38% of the test corpus's true authors were correctly attributed with R1, 55% with R2, 69% of texts are truly attributed to the top three (R3), 78.4% to the top four (R4), and 85.2% with R5. Based on the results, it is clear that for large datasets, increasing the number of patterns will help better describe the writing style of authors and improve the results. Also, it has been seen that skip-grams of length between 1 and 3 are more sufficient than the large skip-grams for the attribution task.

When the results from PoS skip-grams, PoS bigrams, and PoS trigrams for this dataset are compared, it is observed that skip-grams do better than bigrams. In Table 7, the results for PoS bigrams and PoS trigrams are shown. The best results are obtained from trigrams for k = 200, where the success ratio of identifying the true authors over the test corpus are 43.4% for R1, 60.6% for R2, 70.8% for R3, 80.6% for R4, and 87.4% for R5.

5. CONCLUSION

In this paper, a Part-of-Speech Skip-Gram (PoSSG)-based method for authorship attribution is presented. A sequential pattern mining approach is used for PoSSG pattern extraction. Jaccard's Coefficient and Sorensen-Dice were used for computing the similarity of two documents. To test the effectiveness of the proposed methodology, numerous experiments were carried out on two datasets with a variety of parameters. The study demonstrates that the presented AAPoSSG algorithm performs better than conventional character-n-gram-based techniques. According to the results, the proposed approach performs better on larger documents than on smaller ones since authors' individual writing styles are more clearly reflected in larger texts. In this study, the dataset for Canadian authors' novels with large documents produced better results when the number of PoS patterns was set to 100, whereas the dataset for C10-Attribution with short documents produced good results when the number of PoS patterns was set to 200. A future study can examine how well the proposed work performs with more authors and cross-domain datasets.

REFERENCES

- [1] Stamatatos, E., Fakotakis, N., Kokkinakis, G. (1999). Automatic authorship attribution. Ninth conference of the European Chapter of the Association for Computational Linguistics, p. 158-164, Morristown, NJ: Association for Computational Linguistics. <https://aclanthology.org/E99-1021>, accessed on Jan. 10, 2023.

- [2] Zheng, R., Qin, Y., Huang, Z., Chen, H. (2003) Authorship analysis in cybercrime investigation. *International conference on intelligence and security informatics*. Springer pp. 59-73. https://doi.org/10.1007/3-540-44853-5_5
- [3] Juaola, P. (2006). Authorship attribution. *Foundations and Trends. Information RETRIEVAL journal*, 1(3): 233-334. <https://doi.org/10.1561/1500000005>
- [4] Houvardas, J., Stamatatos, E. (2006) N-gram feature selection for authorship identification. *International conference on artificial intelligence: Methodology, systems, and applications*. Springer, pp. 77-86. https://doi.org/10.1007/11861461_10
- [5] van Halteren, H. (2007). Author verification by linguistic profiling: An exploration of the parameter space. *ACM Transactions on Speech and Language Processing*, 4(1): 1-17. <https://doi.org/10.1145/1217098.1217099>
- [6] Stamatatos, E. (2009). A survey of modern authorship attribution methods. *Journal of the American Society for Information Science and Technology*, 60(3): 238-556. <https://doi.org/10.1002/asi.21001>
- [7] Koppel, M., Schler, J., Argamon, S. (2009). Computational methods in authorship attribution. *Journal of the American Society for Information Science and Technology*, 60(1): 926. <https://doi.org/10.1002/asi.20961>
- [8] Sidorov, G., Velasquez, F., Stamatatos, E., Gelbukh, A., Chanona-Hernandez, L. (2014) Syntactic n-grams as machine learning features for natural language processing. *Expert Systems with Applications*, 41(3): 853-860. <https://doi.org/10.1016/j.eswa.2013.08.015>
- [9] Markov, I., Stamatatos, E., Sidorov, G. (2018) Improving cross-topic authorship attribution: The role of pre-processing. *Computational Linguistics and Intelligent Text Processing*, 10762: 289-302. https://doi.org/10.1007/978-3-319-77116-8_21
- [10] Mitra, B., Craswell, N. (2018) An introduction to neural information retrieval. *Now Foundations and Trends*, 13(1): 1-129. <https://doi.org/10.1561/15000000061>
- [11] Rexha, A., Kroll, M., Ziak, H., Kern, R. (2018) Authorship identification of documents with high content similarity. *Scientometrics*, 115(1): 223-237. <https://doi.org/10.1007/s11192-018-2661-6>
- [12] Zhang, H., Nie, P., Wen, Y., Yuan, X. (2018) Authorship Attribution for Short Texts with Author-Document Topic Model. *11th International Conference, KSEM 2018 Proceedings, Part I*, 11061. https://doi.org/10.1007/978-3-319-99365-2_3
- [13] Bacciu, A., La Morgia, M., Mei, A., Nemmi, E.N., Neri, V., Stefa, J. (2019). Cross-domain authorship attribution combining instance-based and profile-based features notebook for PAN at CLEF 2019. *CEUR Workshop Proc.*, 2380: 9-12.
- [14] Barlas, G., Stamatatos, E. (2020) Cross-domain authorship attribution using pre-trained language models. *Artificial Intelligence Applications and Innovations*, 583.
- [15] Fabien, M., Villatoro-Tello, E., Motliceck, P., Parida, S. (2020) BertAA: BERT fine-tuning for Authorship Attribution. *17th International Conference on Natural Language Processing*, Patna, India, pp. 127-137. <https://aclanthology.org/2020.icon-main.16>, accessed on Dec. 12, 2022.
- [16] Devlin, J., Chang, M.W., Lee, K., Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *Association for Computational Linguistics*, 1: 4171-4186. <http://dx.doi.org/10.18653/v1/N19-1423>
- [17] Boughaci, D., Benmesbah, M., Zebiri, A. (2019) An improved N-grams based Model for Authorship Attribution. *International Conference on Computer and Information Sciences (ICCIS)*, Sakaka, Saudi Arabia, pp. 1-6. <https://doi.org/10.1109/ICCISci.2019.8716391>
- [18] Pizarro, J. (2020). Using N-grams to detect Fake News Spreaders on Twitter. *CLEF*, 2696.
- [19] Khomytska, I., Teslyuk, V. (2020) Statistical models for authorship attribution. *Advances in Intelligent Systems and Computing IV*, 1080: 579-592.
- [20] Custódio, J.E., Paraboni, I. (2021) Stacked authorship attribution of digital texts. *Expert Systems with Applications*, 176: 114866. <https://doi.org/10.1016/j.eswa.2021.114866>
- [21] Abedzadeh, A., Ramezani, R., Fatemi, A. (2021) A Weighted TF-IDF-based Approach for Authorship Attribution. *11th International Conference on Computer Engineering and Knowledge (ICCKE)*, Mashhad, Iran, pp. 188-193. <https://doi.org/10.1109/ICCKE54056.2021.972147>
- [22] Howe, D.C. (2009). Rita: creativity support for computational literature. *Proceedings of the 7th Conference on Creativity & Cognition*, Berkeley, California, 205-210. <https://doi.org/10.1145/1640233.1640265>
- [23] Toutanova, K., Klein, D., Manning, C. D., Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. *North American Chapter of the Association for Computational Linguistics*. <https://doi.org/10.3115/1073445.1073478>
- [24] Fournier-Viger, P., Gomariz, A., Gueniche, T., Mwamikazi, E., Thomas, R. (2013) Tks: Efficient mining of top-k sequential patterns. In *Advanced Data Mining and Applications*, 8346: 109-120. https://doi.org/10.1007/978-3-642-53914-5_10
- [25] <https://www.gutenberg.org/ebooks/>, accessed on Jan. 10, 2023.
- [26] Stamatatos, Efstathios. (2006). Stamatatos06 Author Identification: C10-Attribution [Data set]. In *CLEF 2015 Labs and Workshops, Notebook Papers*. Conference title: PAN at Conference and Labs of the Evaluation Forum 2015 (PAN at CLEF 2015). Zenodo. <https://doi.org/10.5281/zenodo.375906>