Hindawi
Scientific Programming
Volume 2019, Article ID 9431073, 14 pages
https://doi.org/10.1155/2019/9431073

Research Article

Design and Implementation of a Machine Learning-Based Authorship Identification Model

Waheed Anwar,[1] **Imran Sarwar Bajwa**,[1] **and Shabana Ramzan**[2]

[1]*Department of Computer Science & IT, The Islamia University of Bahawalpur, Bahawalpur, Pakistan*
[2]*Department of Computer Science, Govt. Sadiq College Women University, Bahawalpur 63100, Pakistan*

Correspondence should be addressed to Waheed Anwar; waheed@iub.edu.pk

Received 29 October 2018; Accepted 18 December 2018; Published 16 January 2019

Guest Editor: Vicente García-Díaz

In this paper, a novel approach is presented for authorship identification in English and Urdu text using the LDA model with n-grams texts of authors and cosine similarity. The proposed approach uses similarity metrics to identify various learned representations of stylometric features and uses them to identify the writing style of a particular author. The proposed LDA-based approach emphasizes instance-based and profile-based classifications of an author's text. Here, LDA suitably handles high-dimensional and sparse data by allowing more expressive representation of text. The presented approach is an unsupervised computational methodology that can handle the heterogeneity of the dataset, diversity in writing, and the inherent ambiguity of the Urdu language. A large corpus has been used for performance testing of the presented approach. The results of experiments show superiority of the proposed approach over the state-of-the-art representations and other algorithms used for authorship identification. The contributions of the presented work are the use of cosine similarity with n-gram-based LDA topics to measure similarity in vectors of text documents. Achievement of overall 84.52% accuracy on PAN12 datasets and 93.17% accuracy on Urdu news articles without using any labels for authorship identification task is done.

1. Introduction

Stylometry is the study of distinct linguistic styles and individual writing practices with the purpose of determining the authorship of a written piece of text [1]. A writing style represents the linguistic choices of a writer that persist throughout one's work. The stylometric research is inspired by the hypothesis that every person has a unique and distinct writing style, referred to as "stylistic fingerprint" [2] that can be measured and learned. Here, a stylistic fingerprint of a writer means a set of features frequently used by that author such as word length, sentence length, choice of certain words, and syntactic structure of a sentence. The stat-of-the-art perspective of stylometry research is authorship analysis [2–7]. In the recent past, the domain of authorship analysis has embraced new dimensions of research typically with the emergence of machine learning techniques for text mining. One of the recent and emerging trends in authorship analysis is computational extraction of stylometric features from the text of an author instead of engineering the stylometric features manually [8–10]. The main focus of authorship identification is deciding the most probable author of a target document among a list of known author's [3]. From a machine learning aspect, authorship identification can be perceived as one label multiclass text classification problem where the role of classes are played by contestant authors [11].

The detailed literature review in the domain of authorship identification for the last two decades revealed that it is a field of great interest and has been mainly applied on the English language [4, 6, 12–14]. Additionally, few solitary efforts were under taken for other languages: Arabic [6, 15], Dutch [16–18], Greek [7, 19], and Portuguese [20, 21]. However, there is no major contribution in the field of authorship identification of Urdu text except for Urdu poetry [22]. To the best of our knowledge, there is neither a

theoretical model nor a subsequent accurate tool available for authorship identification of Urdu newspaper columns. Therefore, such authorship identification application for Urdu language is timely as discussed in [23]. In this paper, an improved approach is discussed that uses similarity measures as tf-idf along cosine similarity and a KNN-based classification module for more accurate results. This paper also compares the results of our approach with PAN12 dataset.

Latent Dirichlet allocation (LDA) [24] was found to be a flexible generative probabilistic unsupervised topic model typically used for the authorship identification for text documents [8, 9, 25, 26]. LDA was used with similarity measuring techniques such as Hellinger [9]. During the literature review, it has been found that the results of the previously used similarity measuring techniques provide low accuracy and there is a need in improvement in topic matching process of LDA-based author identification. In this paper, we propose a methodology for the use of *n*-grams with LDA to find similarity in vectors of text documents by using cosine distance metric. In the literature review, it was identified that the cosine similarity [27] has not been previously employed with LDA for authorship identification of the text documents. One of the objectives of the research presented in this paper was to investigate the behaviour of cosine similarity with LDA in comparison with other similar previously used techniques for authorship identification.

The presented approach builds the LDA model on *n*-grams texts instead of simple text. *N*-grams have been used to keep personal stylistic attributes of the text writer. The LDA model generates topical representation of text documents. These topical representations have been used to build cosine similarity metric for KNN classifier. Here, LDA's application on *n*-grams words not only keep various stylistic fingerprints to identify the writing style of a particular author but permits us to analyse a large dataset of Urdu newspaper articles and can identify the potential author for testing dataset. The presented approach emphasizes on author instance-based and profile-based classifications of text. We used LDA which can handle high-dimensional and sparse data, allowing more expressive representation of texts. LDA is also suitable considering the heterogeneity of the dataset, inherit ambiguity of Urdu language text, and diversity in writing styles of authors. A large dataset was collected for performance testing of the presented approach. The results of experiments show superiority of the proposed approach over the state-of-the-art representations and other algorithms used for authorship identification. The contributions of the presented work are the use of KNN classifier with cosine similarity metric extracted from LDA topics to measure similarity in vectors of text documents and achievement of satisfactory results on English and Urdu news articles without using any labels for authorship identification task.

The rest of the paper is structured as follows: Section 2 discusses the outcomes of the detailed literature. Section 3 describes the materials and methods of the presented research and the LDA-based used approach for authorship identification in PAN12 authorship identification task and Urdu newspaper articles; Section 4 provides details of the experiments, their results, and discussions; at last in Section 5, conclusion are drawn.

2. Related Work

In the literature, a large number of works in the past had been focused on computational linguistics-based methods for identification of stylometric features from text and their application to attribute possible author of the text. The focus of these approaches was to improve various tasks of authorship analysis of a piece of text such as authorship identification, author verification, and author profiling.

The first approach to authorship identification is the use of univariate or multivariate measures that can reflect the style of a particular author. Individual measures were proposed such as word occurrence or frequencies of specific word [28], mean sentence length or wge word length [29], and word richness [11]; however, none of these univariate measures prove to be adequate [30]. The idea behind the multivariate approach is to take documents as points in vector space, and by using some suitable similarity measures, assign the query document to the author, whose documents are closest to the query document [31]; furthermore, other distance-based similarity metrics such as Euclidean distance, Kullback–Leibler, and Hellinger distance were applied to various feature sets for authorship identification [4, 19, 22].

The second approach is statistical machine learning techniques. Individual author is a category value, and a classification model is built. Machine learning techniques are further separated into two subgroups, one is supervised and other is unsupervised. In supervised learning, a classifier is built using both features and the categorical value. However, unsupervised models work on unlabelled data [24]. For authorship identification, supervised techniques include support vector machine (SVM) [13, 32, 33], decision trees [6], linear discriminant analysis [34], and neural networks [35, 36]. The support vector machine outperformed other supervised techniques such as linear discriminant analysis and neural networks in terms of accuracy. Unsupervised classification techniques include principal component analysis (PCA) [37, 38], cluster analysis [39], word2vec [40], doc2vec or distributed document representation [41], and LDA [8, 9, 25, 26]. The work discussed in [23] is the first attempt to address author identification in Urdu text and that approach is improved in this paper by using tf-idf along cosine similarity and a KNN-based classification module for more accurate results.

The first systematic study of authorship identification by using enhanced version of LDA was presented by Michal Rosen-Zvi et al. [8]. The LDA model has the ability to identify all hidden topics from large numbers of features and present them as LDA topics, thus, serving for dimensionality reduction and making it attractive for text analysis problems.

We collected articles from Web of Science by applying the search query "authorship identification" in titles. The query provided 714 articles with default settings in the span of 2007 to 2018 as now we cannot get articles beyond 2007 from Web of Science. We used CiteSpace tool

(URL http://cluster.ischool.drexel.edu/~cchen/citespace/download/) to visualize patterns and trends in the authorship attribution domain. Figure 1 shows most influential authors with cited reference network.

3. Materials and Methods

In order to make the result of the present study reproducible, in this section, the main steps of our proposed framework for authorship identification are discussed; that is, English and Urdu corpus, their datasets, text preprocessing, models with their parameter settings, and experiments. The materials used are corpora in English (Table 1) and Urdu (Table 2), and datasets (Tables 3 and 4) have been generated from these corpora and the most important have been *n*-grams-based inferred topics from LDA. The relevant methods include the methods of preprocessing, various features extraction and selection, document term matrix preparation, topics extraction using latent Dirichlet allocation, and the cosine metric for KNN classifier-based methodology for classification.

We used a publicly available dataset in English from the authorship identification competition of the PAN 2012 [42]. The competition included six tasks for authorship identification for both close and open classes and two tasks for intrinsic plagiarism. Close class means the author of a test or an anonymous text is among the closed set of candidate authors of training data, and open class means the author of a test document might be none of the closed set of candidates. The task notation and description are listed in Table 1. The PAN12 dataset has training and testing parts for closed-class and open-class authorship identification problems.

In the PAN 2012 competition, the training data were extremely small with only two documents per author provided for training. The length of documents varies: in tasks A to D, short documents were given having words in range 2,000 to 13,000 words, while tasks I and J dealt with long documents containing approximately 30,000 to 160,000 words. Tasks E and F were related to plagiarism detection and are out of scope for the present study. From the PAN12 competition, there were two types of tasks on the bases of testing documents: the first one is author-dependent recognition where all the authors of the testing documents were among the training documents authors and the second one is author-independent recognition where some testing documents were from unknown authors which were not part of training documents authors. We only selected author-dependent tasks (A, C, and I); however, author-independent tasks were not in scope of the present study.

The UrduCorpus has 4,800 documents written by twelve well-known Urdu newspaper columnists with 400 articles per author. It contains 5,631,850 words (tokens) in total; at the document level, the mean length was 1174 words. The longest document was written by Dr Muhammad Ajmal Niazi (2,170 words) and the shortest by Irshad Ansari (86 words). When considering the mean length per author, Irshad Ansari wrote the shortest document (396 words per document), while Javed Chaudhary is the author of the longest document (1,690 words per document).

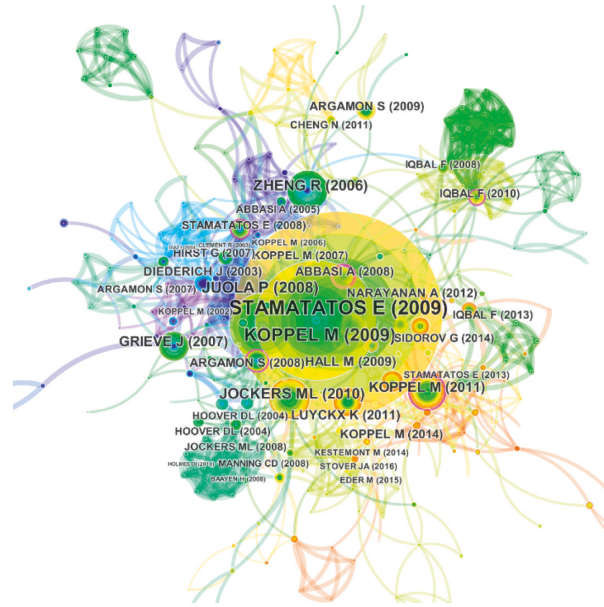


Figure 1: Network of most influential authors in the authorship identification domain.

3.1. Datasets. In the preparation of the datasets from PAN12 and UrduCorpus, we used two representations of author-specific documents.

3.1.1. Instance-Based Dataset. In this type of representation, all documents were treated individually.

3.1.2. Profile-Based Dataset. All the author-specific documents were concatenated into a single file. This single document represents an individual author.

We prepared 12 datasets from PAN12 as shown in Tables 2 and 3; among these datasets, six were instance-based as original text and *n*-grams representation of text and six were profile-based as original text and *n*-grams representation of text.

The number of training and testing documents in the profile-based datasets are equal, whereas in the instance-based datasets, training documents are double than testing for model evaluation.

Similarly, we prepared four datasets from UrduCorpus as shown in Table 3. In Table 3, among these datasets, two were instance-based as original text and *n*-grams representation of text and two were profile-based as original text and *n*-grams representation of text. We randomly used 75% and 25% of data by each author for training and testing, respectively.

Two profile-based datasets of UrduCorpus have only twelve lengthy documents for training, and each dataset has equally twelve hundred test documents for model evaluation.

Figure 2 depicts the proposed framework for authorship attribution using the topic modelled with LDA with cosine metric for the KNN classifier.

Table 1: PAN 2012 authorship identification competition tasks.

Task name	Type	Authors	Descriptions
A	Authorship identification	3	Closed class, short text
B	Authorship identification	3	Open class (of task A), short texts
C	Authorship identification	8	Closed class, short texts
D	Authorship identification	8	Open class (of task C), short texts
I	Authorship identification	14	Closed class, novel length texts
J	Authorship identification	14	Open class (of task I), novel length texts
E	Intrinsic plagiarism	2–4	Mixed set of paragraphs by individual author
F	Intrinsic plagiarism	2	Consecutive intrusive paragraphs by individual author

Table 2: PAN12 datasets used in the experiments.

Dataset	Description	Training documents	Testing documents
A_1	Instance-based (original text)	6	6
A_2	Instance-based (n-grams)	6	6
A_3	Profile-based (original text)	3	6
A_4	Profile-based (n-grams)	3	6
C_1	Instance-based (original text)	16	8
C_2	Instance-based (n-grams)	16	8
C_3	Profile-based (original text)	8	8
C_4	Profile-based (n-grams)	8	8
I_1	Instance-based (original text)	28	14
I_2	Instance-based (n-grams)	28	14
I_3	Profile-based (original text)	14	14
I_4	Profile-based (n-grams)	14	14

Table 3: Urdu datasets used in the experiments.

Dataset	Description	Training documents	Testing documents
D_1	Instance-based (original text)	3600	1200
D_2	Instance-based (n-grams)	3600	1200
D_3	Profile-based (original text)	12	1200
D_4	Profile-based (n-grams)	12	1200

3.2. Document Preprocessing. It is observed from the literature review that it is not needed for vigorous preprocessing in authorship attribution. As writer's grammatical mistakes, their preferences of letter abbreviation, letter capitalization, and word prefixes and suffixes all are essential part of one's writing style. In this case, it is not feasible to correct grammatical mistakes or stem words, such actions may reduce the number of features specific to writer.

3.2.1. Tokenization. Tokenizing means to change sentences into small units like words and characters. We used Natural Language Toolkit (NLTK) [43] for tokenizing at word-level after ignoring all whitespaces.

3.2.2. Lowercasing. Languages such as English have uppercase and lowercase texts. It is recommended to lowercase them before any further preprocessing. We applied this process on the PAN12 dataset. In the Urdu language, we only have one case, so no need to change it into any other.

3.2.3. N-Grams Generation. N-gram is a grouping of adjacent words or characters of length n. We can generate these n-grams for any language. In other words, n-grams features are language independent. They can capture the language structure of a writer; for instance, what character or word was anticipated to follow the given one. The choice of n is very vital in n-grams generation. If the value of n is small which produces short n-grams, we may fail to capture important differences. On the contrary, if the value of n is large, it produces long n-grams; as a result, we may only stick to specific cases. Ideal n-grams length really depends on the application, a good rule of thumb in word level n-grams is to use n-grams where $n \in \{1, \ldots, 5\}$. To overcome the limitation of the bag of the word model where the contextual information is lost, we can capture more semantically meaningful information from text with n-grams. Lexical n-grams are popular, as they have shown to be more effective than character n-grams [44] and syntactic n-grams when all the possible n-grams are used as features [45]. Moreover, it has been shown to be effective in identifying the gender of tweeters [46]. For ease of understanding, we used underscores (_) to replace spaces in

Table 4: *N*-grams (1–5) for sentence "writing is footprint of a writer."

N-grams types	Sentence representation
Word unigrams	Writing, is, footprint, of, a, writer
Word bigrams	Writing_is, is_footprint, footprint_of, of_a, a_writer
Word trigrams	Writing_is_footprint, is_footprint_of, footprint_of_a, of_a_writer
Word fourgrams	Writing_is_footprint_of, is_footprint_of_a, footprint_of_a_writer
Word fivegrams	Writing_is_footprint_of_a, is_footprint_of_a_writer

word *n*-grams and represent them as a single word in the vocabulary and subsequently in the bag of the word model. Table 4 shows a simple sentence and its complete lists of unigrams, bigrams, trigrams, fourgrams, and fivegrams words generated from it.

For word-level *n*-grams feature vector length varies as choice of *n* varies, it can increase rapidly almost *n*-times with *n*-grams.

3.2.4. Stop Word Removal. Stop words are common words in a given language which has high-frequency in the text of language. For instance, in English words like a, an, the, this, and for are stop words. Stop words generally have minor lexical content, and they have enormous presence in the text document. However, they fail to distinguish it from other texts. Sometimes, we want to remove these words from the document before further processing. In languages such as English, we have stop words list; however, in the Urdu language, we do not have such list. We add constraint that each selected word should not appear in every document. Thus, we want to ignore stop words appearing in almost every document. We ignore all words occurring in 70 percent or more documents. Taking into account this constrain, we ignore 666 most frequent words.

3.2.5. Stemming. Stemming is the process of extracting the base word from the given word. This base word is called the stem or root word. We used a rule-based stemmer with the help of Stanford coreNLP tools to stem datasets words.

3.3. Syntax Analyzer and Feature Extraction. Extracting numerical information from raw text documents is normally termed as feature extraction process. Among extracted features, only those features are selected that best fit the training model. After this process, if the features set dimensionality is huge and difficult for computation, then it requires dimensionality reduction algorithms for appropriate performance. The following feature extracting techniques were used for the proposed model.

3.3.1. TF-IDF. We can produce distinct feature vectors based on information captured from the texts. This could be simply raw frequency of each word or term frequency and inverse document frequency (tf-idf). We can use tf-idf to

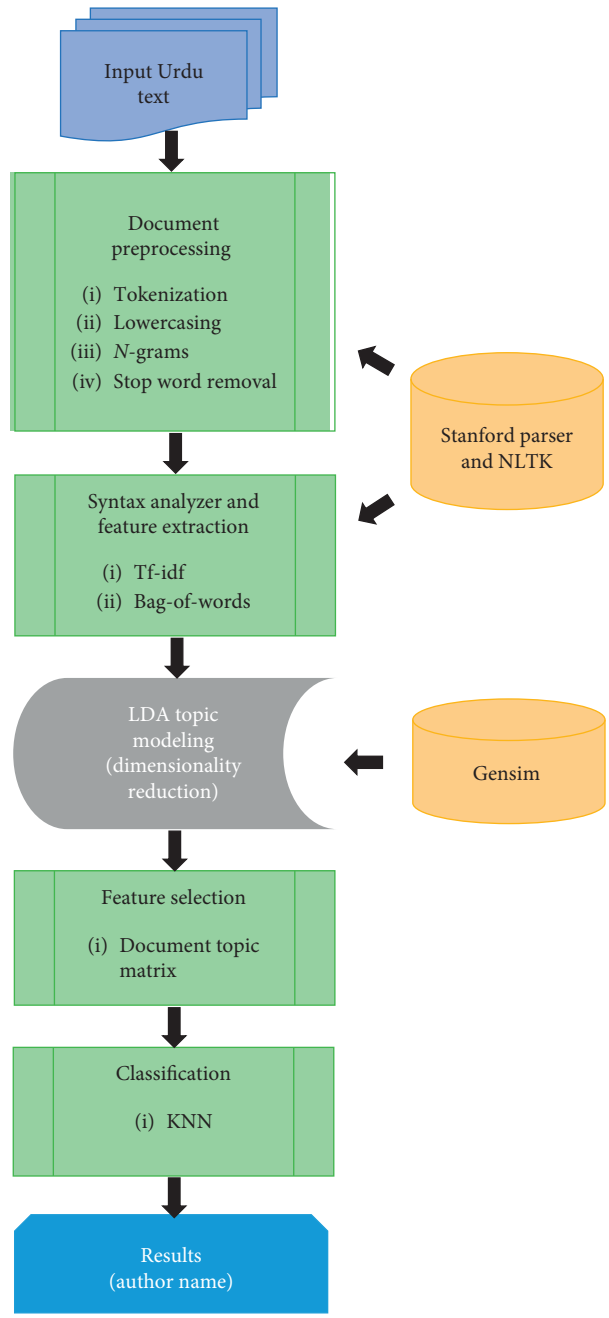


Figure 2: Architecture of text-based forensic analysis approach.

assess how significant a word is in identifying the actual author for a given document. This tf-idf value can be obtained by multiplying the ratio of the word in a document to the reciprocal of the ratio used in the all documents.

3.3.2. Bag of Words Extraction. In natural language processing, bag-of-words is a classic model. In this model, text is considered as a set of words each one having a frequency of occurrence in the corpus; however, their contextual information is lost. In other words, it is order less document features representation in the form of frequencies that occurs in the document to form a dictionary, and this

dictionary may consist of character, character n-grams, words, words n-grams, or some other features extracted from text. If we use all distinct words for our vocabulary, it can increase overall corpus dimensionality which is difficult for computation. For feature selection, we have applied two schemes.

(1) Term Document Frequency. We have been considering only those terms having occurrence in 10 or more documents (tdf $\geq$ 10), and it reduced vocabulary size to 104,867 terms in instance-based n-grams dataset of UrduCorpus.

(2) Percentage of Documents for Term. As a second constrain, we wanted to remove stop words or other most frequent words from corpus. We ignore all words occurring in 90 percent or more documents. Taking into account this second constrain, we ignore 666 most frequent words.

Finally, we obtain a vocabulary of size 104,201 terms in instance-based n-grams dataset D_2, similar feature selection schemes were applied on simple instance and profile-based datasets we obtained vocabulary of size 44,634 terms and for profile-based n-grams dataset applying slightly different feature selection schemes as there has been only twelve long concatenated documents. We have selected only those terms which occurred in two or more documents, however, not more than ten documents. We capture 55,423 terms for vocabulary.

For PAN12 English datasets, training documents words and distinct words for each dataset are given in Table 5.

We applied different feature selection techniques as the training documents for each author were only two in each dataset. First, we selected topmost frequent 2,500 words for each author, and then ignoring the common words which other authors also used, we only selected author specific words. In this way, stop words were ignored and only those words were captured with which were author specific. We also add the second constraint that, among these author specific words, only top 300 most frequent words for each author were selected to build balance vocabularies for A1, A_3, C_1, C_3, I_1, and I_3 datasets. For datasets A_2, A_4, C_2, C_4, I_2, and I_4 having n-grams words, we selected top 500 most frequent words for each author to build vocabularies of equal size with respect to authors.

3.4. Document Term Matrix. Text documents are generally represented as a vector, where each attribute represents particular term frequency occurrence. This vector form representation can be used to find the similarity between the two corresponding documents. We prepared document term matrix (Figure 3) from training dataset based on the selected features which were saved in the form of vocabulary by using Gensim dictionary class. The LDA model looks for repeating term patterns in the entire document term matrix.

3.5. Feature Selection Using LDA. We can use topic models for the purpose of information retrieval and feature selection from unstructured text. A topic modelling algorithm, for example, latent Dirichlet allocation [24], is useful for organizing large volume of textual data into overlapping clustering of documents [24, 47], which differ from other text mining approaches, which are rule-based and use dictionary or regular expressions-based keyword searching. LDA is a flexible generative probabilistic topic model for collection of discrete data, which expresses the documents as a collection of topics mixture with different probabilities for these topics in documents, and each topic is expressed as a list of words with probabilities for them to belong to that topic.

Table 5: PAN12 datasets used in the experiments.

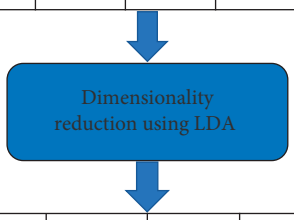
Dataset	Training documents words	Distinct words	Vocabulary size
A_1	25,771	3,252	900
A_2	128,845	48,012	1,500
A_3	25,771	3,252	900
A_4	128,845	48,012	1,500
C_1	96,052	26,654	2,400
C_2	480,250	133,256	4,000
C_3	96,052	26,654	2,400
C_4	480,250	133,256	4,000
I_1	2,353,267	137,315	4,200
I_2	11,766,325	7,839,471	7,000
I_3	2,353,267	137,315	4,200
I_4	11,766,325	7,839,471	7,000

We have used LDA to reduce the dimension of document term matrix into a new matrix (Figure 3). We named new matrix as document topic matrix, as each cell represents specific topic weight in that document and each matrix row now have topical representation of whole document in the normalized form.

This representation achieved dimensionality reduction of the document term matrix. For example, for Urdu dataset, reduction of D_2 document term matrix from $3600 \times 104{,}201$ to document topic matrix 3600×120 is achieved, which is almost 91% less of the vocabulary of the corpus. This result is extremely helpful in features selection and classification of documents.

3.6. Hyper Parameters and Parameters of LDA. LDA has corpus-level parameters named hyperparameters α and β sampled only once, and these parameters are from the Dirichlet distribution. The first parameter α controls the distribution of document topics, and β is accountable for distribution of topic words. The high value of α means each document possibly contains mixture of almost every topic not a particular one, while low value of α means that the document contains some topics. Similarly, high β means that each topic contains a mixture of most of the words not just specific words. The low value of β means the topic may represent a blend of just some of the words. In a nutshell, high α will produce documents more identical to each other and high β will produce topics more identical to each other. However, the number of topics k is user defined, and we need to figure out the number of topics based on the data. Thus, each document d_i, for $i = 1, \ldots, n$, is generated based

	Term1	Term2	Term3	…	Term104201
Doc1	2	0	1	…	0
Doc2	3	3	0	…	1
Doc3	2	1	4	…	1
⋮	⋮	⋮	⋮	⋱	⋮
Doc3600	1	0	2	…	0



	Topic1	Topic2	Topic3	…	Topic120
Doc1	0.0630	0.0318	0.0930	…	0.0630
Doc2	0.1392	0.0000	0.0426	…	0.1392
Doc3	0.0000	0.0165	0.0515	…	0.0010
⋮	⋮	⋮	⋮	⋱	⋮
Doc3600	0.0007	0.0005	0.0000	…	0.0515

Figure 3: Conversion of the document term matrix to the document topic matrix.

on a distribution over the k topics, where k defines the maximum number of topics. This value is fixed and defined a priori; a lower value for the number of topics may result in border topics such as education, sports, and fashion, and a larger value for the number of topics may result in more focused topics such as science, football, and hairstyle. A large k value for topics means that the algorithm requires lengthier passes to estimate the word distribution for all the topics, and a good rule of thumb is to choose a value that makes sense for a particular case; in the present context, we may consider $k = 12$ at least assuming that each k value corresponds to the individual author writing style and thus choosing $k = 12$ was a sensible choice, and a larger k value than 12 was required to imitate the versatility of the writing style of a particular author as two or more topic distributions were more helpful in this regard. However a lower k value than 12 does not make any sense. We used k between 12 and 120 with the interval of 10.

3.7. LDA + Cosine Similarity. This method is our main contribution, as it achieves state-of-the-art performance in authorship identification with many candidate authors. The main idea of our approach is to use the LDA model in such a way that it provides us dimensionality reduction along with maintaining the author specific writer style and then use cosine similarity in LDA model topic space, to determine the most likely author of the test document. We used n-grams to capture the author writing style. Documents were represented as bag-of-words, so each document from both training and test sets converted into sparse vector and were mapped into LDA topic space to generate a vector representation for each one, which can be represented as u_i and v_i as outcomes, respectively.

In text similarity measures, cosine similarity is one of the most popular one. It is a distance metric from computational linguistics to measure similarity between document vectors. In order to find cosine similarity between two documents u and v, first we need to normalize them to one in L_2 norm:

$$\sum_{i=1}^{k} u_i^2 = 1. \tag{1}$$

Now, cosine similarity between these two normalized vectors u and v will be the dot product of them:

$$\cos(u, v) = \frac{\sum_{i=1}^{k} u_i v_i}{\sqrt{\left(\sum_{i=1}^{k} u_i^2\right)}\sqrt{\left(\sum_{i=1}^{k} v_i^2\right)}}, \tag{2}$$

where u_i and v_i are the vectors of n dimensions over the document sets $\mathbf{u}$ and $\mathbf{v}$ where $i = 1, 2, 3, \ldots, k$.

Cosine similarity is considered as the one of the best in similarity measurement. Cosine similarity is very simple in implementation complexity as in Gensim [48]. We also used different evaluation metrics in order to validate and compare our results.

3.8. Classification. Text documents are generally represented as a vector where in a document each attribute represents particular term frequency occurrence. This term vector form representation is used to find the similarity between the two corresponding documents. We can apply KNN to our data that will learn to classify new articles based on their distance to our known articles (and their labels). The algorithm needs a distance metric such as Euclidian distance or cosine similarity to determine which of the known articles are closest to the new one. We used cosine similarity.

4. Results

In our experiments, we validated the proposed authorship identification scheme by performing tests on twelve datasets of PAN12 and four datasets of UrduCorpus. In order to build the low-dimensionality topical representation, the LDA model receives tokenized text documents with n-grams of the training set without any label (without the author to which they belong) as input data type and for evaluation the unlabeled text documents from the testing set. The experiments comprised in testing a cosine base classifier with the

output of LDA k-topics in the corpus, and these topics form a lower-dimensional representation of the corresponding training set based on vocabulary and then evaluating the classifier with the testing set using the same lower-dimensional representation. The overall authorship identification accuracy rate (AR) is computed by the following equation:

$$\text{accuracy rate (AR)} = \frac{\text{number of correctly identified articles}}{\text{total number of test set articles}} \times 100. \tag{3}$$

4.1. Experimental Setup. All the experiments were performed to test the performance and accuracy of the proposed approach using Intel i7 @ 2.8 GHz, operating on windows 10 pro 64-bit with 6 GB memory. Python 3 (Python Software Foundation, Wilmington, DE, USA), NLTK [43], and LDA implementation in Gensim [48] library have been used for the development of the system. LDA implementation in Gensim allows both estimation of topics distribution on training data and inference of these topics on the test data. We used UrduCorpus dataset (Table 3) that belongs to the news domain. Note that change of newspaper may affect the writing style of an author, and similarly over the passage of time, the individual writing style may also change. The nature of articles (their topics) also influences the choice of words; however, every individual has his/her own vocabulary, and he/she may like to use specific words unintentionally which can be used for his/her writing style identification.

To evaluate and compare LDA for authorship identification, we used PAN12 datasets from small datasets having 3 authors and 12 documents, medium datasets having 8 authors and 24 documents, and large datasets of novel length documents with 14 authors and 42 documents and for Urdu dataset, having 4,800 documents written by twelve well-known Urdu newspaper columnists. We used various performance metrics (precision, recall, and F_1-measure) along with accuracy to demonstrate the quality of autodecision-making of cosine-based KNN classifier on PAN12 and UrduCorpus.

4.2. Results and Discussion. In order to validate the results, we evaluated LDA-based authorship identification approach on instance and profile-based datasets with and without n-grams, and we carried out a series of experiments on each dataset with several filters on the term frequency and frequent words removal to generate vocabulary with most appropriate features and different number of LDA topics (12, 24, 36, . . ., 120) for UrduCorpus and LDA topics 3 to 54 for different datasets of PAN12. We presented each experiment with best performance parameter setting for PAN12 as shown in Table 6.

Our results on PAN12 datasets depicts that LDA with n-grams performed better as compare to simple text. When we compared instance-based n-grams with profile-based n-grams, the results were the same, as we have used identical vocabulary in each comparative dataset. Secondly, here we have used balanced number of documents and also the features extracted from these documents were also equal, therefore, in most of the cases, instance-based results were the same as compared to relevant profile-based results. Overall best performance on A, C, and I datasets was 84.53%.

For Urdu datasets, we reported experiments with best performance parameter setting in Table 7.

Our results clearly show that LDA instance-based n-grams approach outperforms LDA profile-based approach significantly, although we were hoping vice versa as mentioned in the literature [9]. In profile-based approach when documents were concatenated into single file to form the author profile, some important authorship features lose their prominence in the profile, and these features have significant discriminating power that sharply contrasts documents between the authors. Secondly, although we have used balanced corpus in terms of the number of documents, the average document length per author varies, so when concatenating documents into the author profile, some profiles have a smaller number of words in total as compared to those of others resulting in unbalanced feature extraction, whereas in the instance-based approach, some documents of an author were long enough to become strong candidate of attributed document. Thirdly, in instance-based approach, different features can be combined easily, whereas in profile-based approach, it is difficult to do so.

We have reported the accuracy percentage yielded by the LDA + cosine similarity approach, in LDA model, setting the number of topics k between (12, 24, 36, . . . , 120) with various vocabulary sizes. Our result shows that varying the number of topics in the LDA model is critical and it has a huge impact on performance. Usually, accuracy increases with the number of topics in a certain range and then begins to decrease. A clear and precise prescription for this parameter is not possible, even in the same dataset with different vocabulary sizes.

In order to evaluate the proposed LDA-based approach on four datasets, we used the same number of topics with identical vocabulary size initially; however, the results were not satisfactory for couple of datasets, as with combination of n-grams document size increases in terms of tokens and length in the dataset, and thus in these datasets, we cannot use the same vocabulary size for each LDA model. We tuned LDA models with different vocabulary sizes keeping the same k topics. We have reported the best performance of each dataset with different vocabulary sizes but the same number of topics between 12 and 120 in the current context, and we may assume that each topic at least matches to the writing style of an author, and thus, fixing $k = 12$ is a reasonable choice. However, the value of k could be larger than 12 representing the fact that each author may require two or more topical representations to well describe the style of a given author. When applying the LDA model on instance-based n-grams dataset with a vocabulary of 104,201 terms and LDA 60 topics, we achieved an accuracy of 93.17% with KNN classifier setting of $k = 7$. Hence, evaluations reported in this graph indicate that the LDA-based authorship

Table 6: Unsupervised classification of documents based on LDA topics with cosine similarity on twelve datasets of pan12.

Dataset with description	Parameters	Accuracy rate (%)
A_1 instance-based (original text)	Vocabulary 900, $k = 6$	83.3
A_2 instance-based (n-grams)	Vocabulary 1500, $k = 6$	**100**
A_3 profile-based (original text)	Vocabulary 900, $k = 3$	83.3
A_4 profile-based (n-grams)	Vocabulary 1500, $k = 3$	**100**
C_1 instance-based (original text)	Vocabulary 2400, $k = 16$	50.0
C_2 instance-based (n-grams)	Vocabulary 4000, $k = 16$	**75.0**
C_3 profile-based (original text)	Vocabulary 2400, $k = 8$	62.5
C_4 profile-based (n-grams)	Vocabulary 4000, $k = 8$	**75.0**
I_1 instance-based (original text)	Vocabulary 4200, $k = 28$	64.3
I_2 instance-based (n-grams)	Vocabulary 7000, $k = 28$	**78.6**
I_3 profile-based (original text)	Vocabulary 4200, $k = 14$	64.3
I_4 profile-based (n-grams)	Vocabulary 7000, $k = 14$	**78.6**

Table 7: Unsupervised classification of documents based on LDA topics with cosine similarity on four datasets of UrduCorpus.

Method and dataset	Parameters	Accuracy rate (%)
LDA instance-based (original text)	Vocabulary 44,634, $k = 24$	91.42
LDA instance-based (n-grams)	Vocabulary 104,201, $k = 60$	**93.17**
LDA profile-based (original text)	Vocabulary 44,634, $k = 60$	91.83
LDA profile-based (n-grams)	Vocabulary 55,423, $k = 72$	91.75

attribution model performs significantly better on instance-based n-grams dataset than other datasets almost on each k topics selection. Note that to further elaborate the results in the following, we used proposed model with instance-based n-grams dataset.

Figure 4 depicts the result of multiple experiments that compare the unsupervised classification of documents based on LDA topics with the KNN classifier on four datasets.

Figure 5 depicts the result of multiple experiments that compare the unsupervised classification of documents based on LDA topics with cosine similarity on PAN12 datasets.

In order to evaluate the proposed LDA-based approach on PAN12 datasets, we used the number of topics k between 3 and 70 depending upon the number of authors and theirs documents with various vocabulary sizes (Table 5). We cannot use the same vocabulary size for each LDA model. We tuned LDA models with different vocabulary sizes keeping dataset specific k topics (Figure 6). We reported the best performance of each dataset with different vocabulary sizes and number of topics between 3 and 70; in the current context, we may assume that each topic at least matches to the writing style of an author thus fixing k equal to 3 for dataset A, 8 for dataset C, and 14 for dataset I is a reasonable choice. However, the value of k could be larger than 3, 8, and 14, respectively, representing the fact that each author may require two or more topical representation to well describe the style of a given author.

When applying the LDA model on A_2 and A_4 datasets with the vocabulary of 1,500 terms and k values of 6 and 3 topics, respectively, we achieved an accuracy of 100% with cosine similarity. On dataset C, we found a best accuracy of 75% with datasets C_2 and C_4 with the vocabulary of 4,000 terms and k values 16 and 8, respectively. Similarly, for I dataset, we found the best accuracy of 78.57% with I_2 and I_4 with vocabulary of 7,000 terms and k values of 28 and 14, respectively. These results clearly indicate that our approach works well both on instance-based and profile-based approaches and on instance-based approach model, and it required a greater number of k topics as compared to the profile-based approach because in profile-based approach, all author specific documents were concatenated and that is why, it required less number of topics. For instance-based approach, the LDA model achieved best results when the k topics were equal to training documents because in PAN12, training documents were limited to only two per author as compared to UrduCorpus where the training documents were 300 per author; therefore, here we assumed that each document represents only one topic. Hence, evaluations reported in these graphs indicate that the LDA-based model performs almost the same on instance-based n-grams dataset and profile-based n-grams dataset however with different k topics. The same Urdu dataset was used to further elaborate the results of the used model with instance-based n-grams datasets.

Figure 6 shows the confusion matrix obtained with proposed methodology on 1200 test documents.

This confusion matrix can be used for various performance measures which can evaluate our results in different ways. As we can see, there is a clear diagonal heatmap which represents the accuracy with respect to the author; however, there were some documents which were misclassified. Three out of twelve authors have at least six misclassified documents towards single author; for instance, ten documents for actual author number eight were misclassified towards predicated author number five which shows some resemblance of one's writing styles. One notable result was that authors with maximum accuracy also did not have any misclassified document in their favor, which shows their unique writing style.

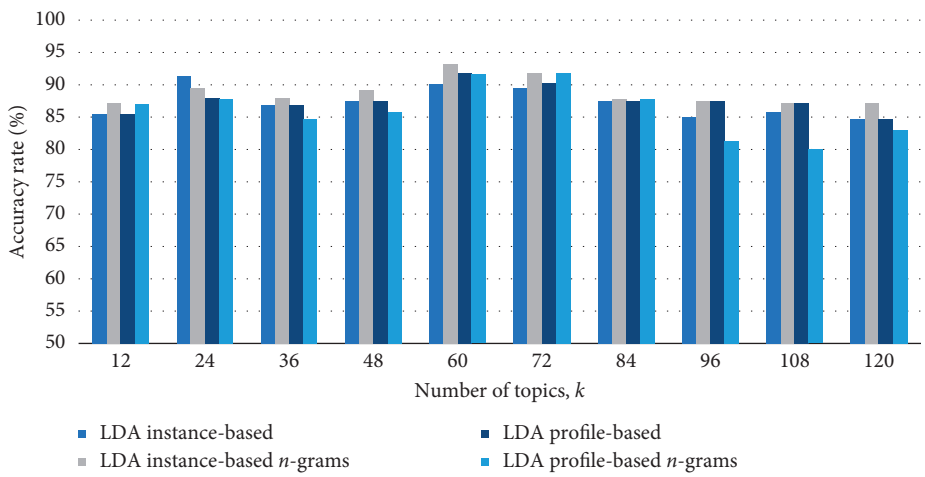


Figure 4: Classification of documents based on LDA topics with KNN classifier on four datasets.

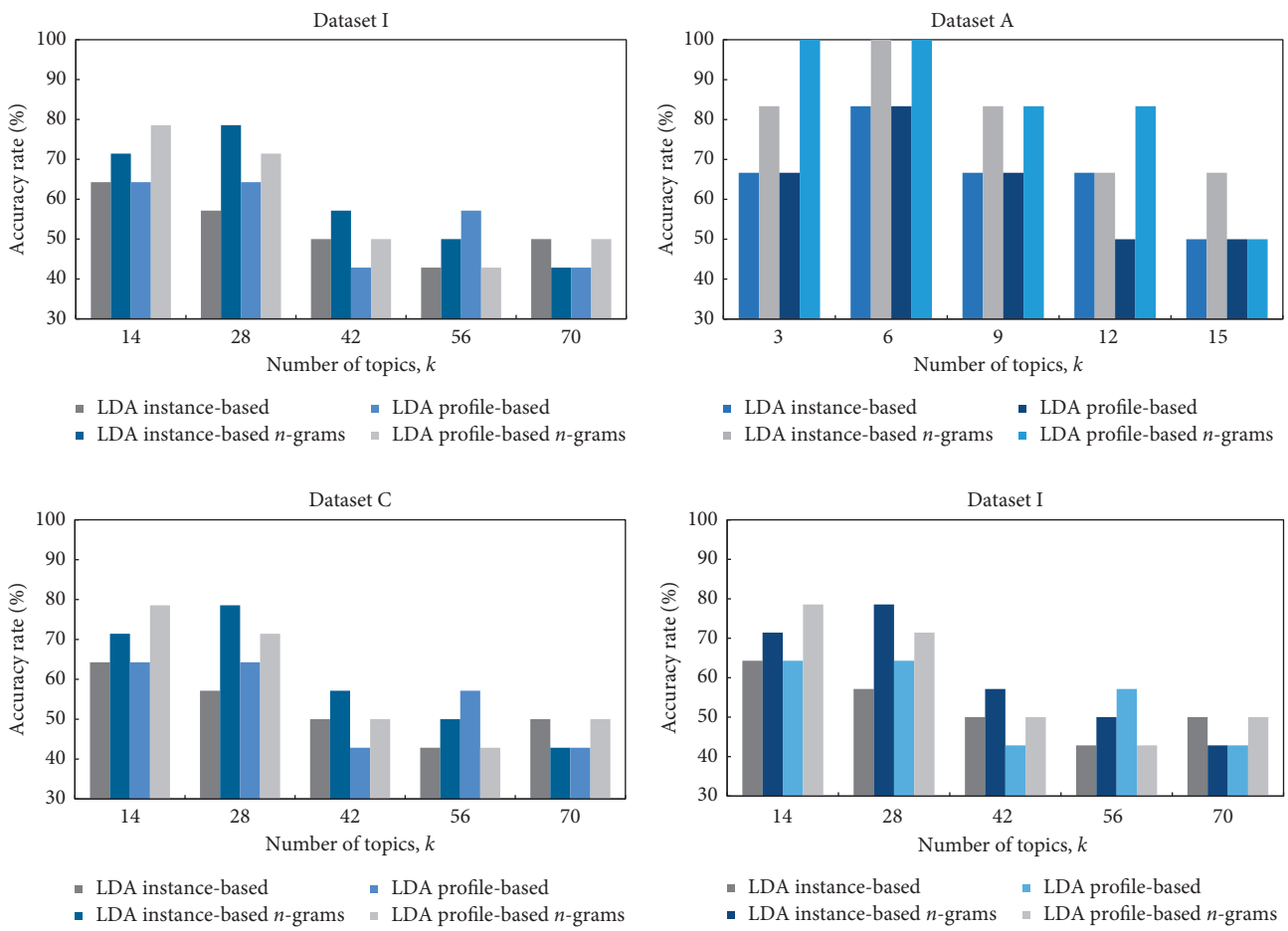


Figure 5: Evaluation of LDA authorship identification using KNN on PAN12 datasets.

Figures 7–9 show the confusion matrix obtained with proposed methodology on PAN12 datasets A, C, and I test documents.

As we can see, from these confusion matrixes of instance-based n-grams datasets A, C, and I of PAN12, there is a clear diagonal heatmap which represents the accuracy with respect to the author; however, there were some documents which were misclassified in C and I datasets. Documents of two out of eight authors were misclassified.

4.3. Interpretation of Misclassified Articles. There can be a number of reasons for misclassification of articles. Firstly, we

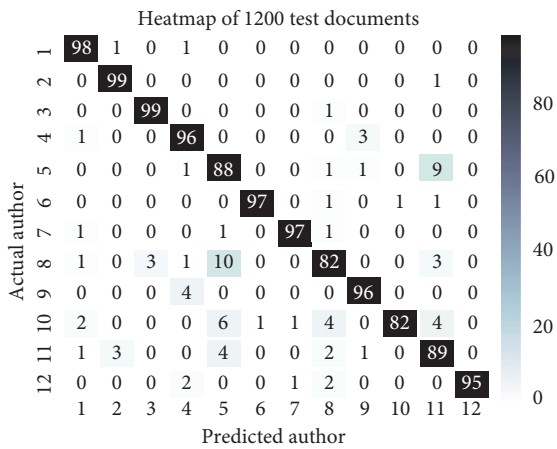


Figure 6: Confusion matrix for test documents of UrduCorpus using instance-based n-grams approach.

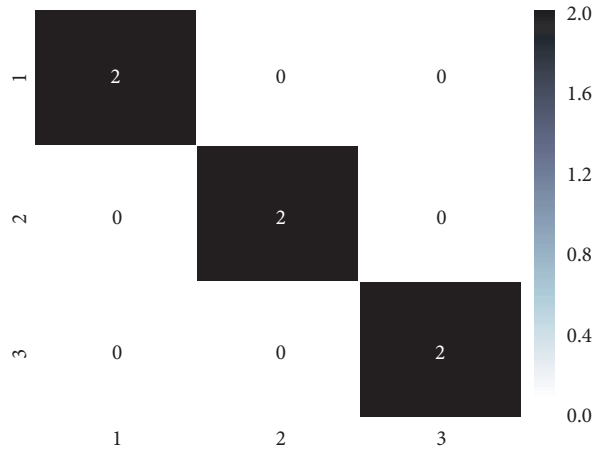


Figure 7: Confusion matrix for test documents of A dataset using instance-based n-grams approach.

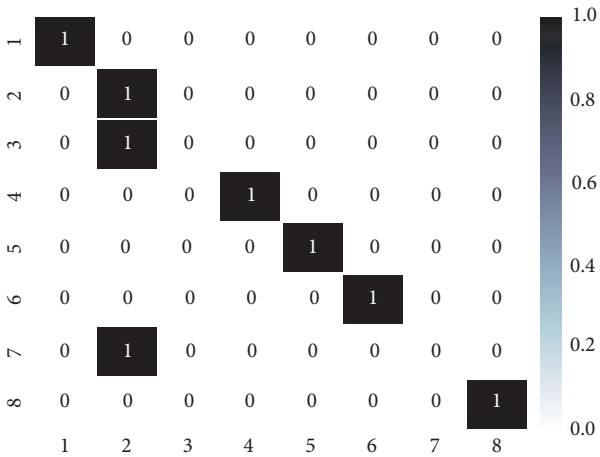


Figure 8: Confusion matrix for test documents of C dataset using instance-based n-grams approach.

found that few authors have the writing style such that, in their articles, first they gave quoted paragraphs of other authors and then discussed their point of view on that topic.

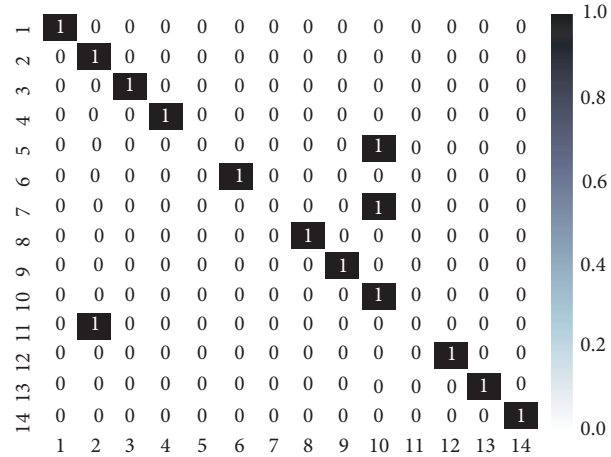


Figure 9: Confusion matrix for test documents of I dataset using instance-based n-grams approach.

In this way, they intermingle their writing style with other authors. Secondly, authors wrote on various domains like politics, religion, sports, and entertainment as the corpus was not domain specific.Our proposed scheme may model an author in consequence of a document in respect to any other author in the specific domain that may result in misclassification. Thirdly, short size of the testing article may be the cause of misclassification.

In Table 8, we reported individual class results in terms of precision, recall, F_1-measure, and accuracy rate obtained on instance-based n-grams dataset by applying the proposed scheme for authorship identification.

The experiment shows our approach models the authors more accurately on n-grams instance-based dataset. We achieved 93.17% accuracy rate on this dataset, and other performance measures were also satisfactory, as precision measure was fluctuating from 81% to 100% and recall measure was between 82% and 99% on individual basis of this dataset. As there is a tradeoff between precision and recall, we attained 93.1% precision and 92.9% recall on 1200 test documents of the instance-based n-grams dataset.

In Tables 9–11, we reported individual class results in terms of precision, recall, F_1-measure, and accuracy rate (percentage of correct answers) obtained on instance-based n-grams datasets A, C, and I of PAN12 by applying the proposed scheme for authorship identification.

We achieved 84.52% of an average accuracy rate on PAN12 instance-based and n-grams datasets A_2, C_2, and I_2, and other performance measures were also satisfactory, as average precision measure was 80%, recall measure was 84.67%, and F_1-measure was 80.67% on these datasets.

5. Conclusions

In this paper, we designated the authorship identification problem in Urdu news articles and English PAN12 tasks in the context of the closed-class problem. As a new authorship identification scheme, we proposed an approach using latent Dirichlet allocation (LDA) paradigm in conjunction with n-

Table 8: Unsupervised classification of author documents on instance-based n-grams Urdu dataset.

Authors	Precision	Recall	F_1 measure
Asad Ullah Ghalib	0.94	0.98	0.96
Dr. Muhammad Ajmal Niazi	0.96	0.99	0.98
Dr. Tauseef Ahmad Khan	0.97	0.99	0.98
Haroon Ur Rashid	0.91	0.96	0.94
Irshad Ahmad Arif	0.81	0.88	0.84
Irshad Ansari	0.99	0.97	0.98
Javed Chaudhary	0.98	0.97	0.97
Karnal R Ikram Ullah	0.87	0.82	0.85
Khursheed Nadeem	0.95	0.96	0.96
Nawaz Raza	0.99	0.82	0.90
Nazeer Naji	0.83	0.89	0.86
Qayyum Nizami	1.00	0.95	0.97
Average	0.931	0.929	0.930

Table 9: Unsupervised classification of author documents on instance-based n-grams PAN12 dataset A.

Authors	Precision	Recall	F_1 measure
candidate00001	1	1	1
candidate00002	1	1	1
candidate00003	1	1	1
Average	1.00	1.00	1.00

Table 10: Unsupervised classification of author documents on instance-based n-grams PAN12 dataset C.

Authors	Precision	Recall	F_1 measure
candidate00001	1	1	1
candidate00002	0.33	1	0.5
candidate00003	0	0	0
candidate00004	1	1	1
candidate00005	1	1	1
candidate00006	1	1	1
candidate00007	0	0	0
candidate00008	1	1	1
Average	0.67	0.75	0.69

grams to produce reduced dimension topical representation of documents. We explained how the topical representations of LDA could be used with cosine distance metric for classification of test documents. Our approach yields satisfactory performance. The best result in terms of accuracy and F_1-measure were achieved with n-grams introduction in the model which captures more stylistic features of an author. The lessons learned were that each language required different configurations at each stage, appropriate selection of the dimensionality of the representation is crucial for authorship identification, and it is possible to significantly improve the accuracy results by fine tuning the size of vocabulary and k topics in LDA.

One possible improvement to the study would be the implementation of the supervised learning model to get good accuracy. This would increase the effort of annotating the corpus. Secondly, we could train the model developed in the study, on a larger set of columnists. One could aim to design and deploy an automated website scraper incorporated with the proposed LDA model to collect other such online articles and create a comprehensive database of all such columnists. By doing so, it could probably help authorship identification on a larger scale.

Table 11: Unsupervised classification of author documents on instance-based n-grams PAN12 dataset I.

Authors	Precision	Recall	F_1 measure
candidate00001	1	1	1
candidate00002	0.5	1	0.67
candidate00003	1	1	1
candidate00004	1	1	1
candidate00005	0	0	0
candidate00006	1	1	1
candidate00007	0	0	0
candidate00008	1	1	1
candidate00009	1	1	1
candidate00010	0.33	1	0.5
candidate00011	0	0	0
candidate00012	1	1	1
candidate00013	1	1	1
candidate00014	1	1	1
Average	0.70	0.79	0.73

Data Availability

The implementation and datasets used in this paper are available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

The authors thank Muhammad Omer for his technical guidelines.

References

[1] D. I. Holmes, "The evolution of stylometry in humanities scholarship," *Literary and Linguistic Computing*, vol. 13, no. 3, pp. 111–117, 1998.

[2] D. I. Holmes, "Authorship attribution," *Computers and the Humanities*, vol. 28, no. 2, pp. 87–106, 1994.

[3] P. Juola, "Authorship attribution," *Foundations and Trends in Information Retrieval*, vol. 1, no. 3, pp. 233–334, 2006.

[4] C. E. Chaski, "Empirical evaluations of language-based author identification techniques," *Forensic Linguistics*, vol. 8, no. 1, pp. 1–65, 2001.

[5] M. Koppel, J. Schler, and S. Argamon, "Authorship attribution in the wild," *Language Resources and Evaluation*, vol. 45, no. 1, pp. 83–94, 2010.

[6] A. Abbasi and H. Hsinchun Chen, "Applying authorship analysis to extremist-group Web forum messages," *IEEE Intelligent Systems*, vol. 20, no. 5, pp. 67–75, 2005.

[7] E. Stamatatos, N. Fakotakis, and G. Kokkinakis, "Text genre detection using common word frequencies," in *Proceedings of the 18th Conference on Computational Linguistics*, vol. 2, p. 808, Saarbrücken, Germany, August 2000.

[8] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth, "The author-topic model for authors and documents," in *Proceedings of 20th Conference on Uncertainty in Artificial Intelligence*, pp. 487–494, Banff, Canada, 2004.

[9] Y. Seroussi, I. Zukerman, and F. Bohnert, "Authorship attribution with latent dirichlet allocation," in *Proceedings of Fifteenth Conference on Computational Natural Language Learning*, pp. 181–189, Portland, OR, USA, 2011.

[10] A. Caliskan-Islam, "Stylometric fingerprints and privacy behavior in textual data," , ProQuest Diss. Thesis, 2015.

[11] F. Sebastiani, "Machine learning in automated text categorization," *ACM Computing Surveys*, vol. 34, no. 1, pp. 1–47, 2002.

[12] D. I. Holmes, M. Robertson, and R. Paez, "Stephen crane and the New-York tribune: a case study in traditional and non-traditional authorship attribution," *Computers and the Humanities*, vol. 35, no. 3, pp. 315–331, 2001.

[13] M. Koppel, J. Schler, and S. Argamon, "Computational methods in authorship attribution," *Journal of the American Society for Information Science and Technology*, vol. 60, no. 1, pp. 9–26, 2009.

[14] M. Kestemont, "Overview of the author identification task at PAN-2018: cross-domain authorship attribution and style change detection," *CEUR Workshop Proceedings*, vol. 2125, 2018.

[15] A. S. Altheneyan and M. E. B. Menai, "Naïve Bayes classifiers for authorship attribution of Arabic texts," *Journal of King Saud University-Computer and Information Sciences*, vol. 26, no. 4, pp. 473–484, 2014.

[16] P. Juola and R. H. Baayen, "A controlled-corpus experiment in authorship identification by cross-entropy," *Literary and Linguistic Computing*, vol. 20, no. 1, pp. 59–67, 2005.

[17] J. Hoorn, S. Frank, W. Kowalczyk, and F. van der Ham, "Neural network identification of poets using letter sequences," *Literary and Linguistic Computing*, vol. 14, no. 3, pp. 311–338, 1999.

[18] P. Maitra, S. Ghosh, and D. Das, "Authorship verification – an approach based on random forest notebook for PAN at CLEF 2015," in *Proceedings of Working Notes for CLEF 2015 Conference*, pp. 1–9, Toulouse, France, September 2015.

[19] V. Kešelj, F. Peng, N. Cercone, and C. Thomas, "*N*-Gram-based Author profiles for authorship attribution," in *Proceedings of the conference Pacific Association for Computational Linguistics*, pp. 255–264, Halifax, NS, Canada, 2003.

[20] D. Pavelec, L. Oliveira, E. Justino, and L. Batista, "Using conjunctions and adverbs for author verification," *Journal of Universal Computer Science*, vol. 14, no. 18, pp. 2967–2981, 2008.

[21] R. Sousa Silva, G. Laboreiro, L. Sarmento, T. Grant, E. Oliveira, and B. Maia, "'twazn me!!! ;(' automatic authorship analysis of micro-blogging messages," in *Proceedings of International Conference on Application of Natural Language to Information Systems*, vol. 6716, pp. 161–168, Salford, UK, 2011.

[22] A. A. Raza, A. Athar, and S. Nadeem, "*N*-gram based authorship attribution in Urdu poetry," in *Proceedings of the Conference on Language & Technology*, pp. 88–93, Poznań, Poland, 2009.

[23] W. Anwar, I. Sarwar Bajwa, M. A. Choudhary, and S. Ramzan, "An empirical study on forensic analysis of Urdu text using LDA based authorship attribution," *IEEE Access*, vol. 6, p. 6600, 2018.

[24] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, no. 3, pp. 993–1022, 2003.

[25] R. Arun, R. Saradha, and V. Suresh, "Stopwords and stylometry: a latent Dirichlet allocation approach," *NIPS Work*, pp. 1–4, 2009.

[26] J. Savoy, "Authorship attribution based on a probabilistic topic model," *Information Processing & Management*, vol. 49, pp. 341–354, 2013.

[27] S. Sohangir and D. Wang, "Document understanding using improved sqrt-cosine similarity," in *Proceedings-IEEE 11th International Conference on Semantic Computing, ICSC*, pp. 278-279, San Diego, CA, USA, 2017.

[28] T. C. Mendenhall, "The characteristic curves of composition," *Science*, vol. 9, pp. 237–246, 1887.

[29] G. Yule, "The statistical study of literary vocabulary," *Modern Language Review*, vol. 39, no. 3, pp. 291–293, 1944.

[30] J. Grieve, "Quantitative authorship attribution: an evaluation of techniques," *Literary and Linguistic Computing*, vol. 22, no. 3, pp. 251–270, 2007.

[31] J. Burrows, "'Delta': a measure of stylistic difference and a guide to likely authorship," *Literary and Linguistic Computing*, vol. 17, no. 3, pp. 267–287, 2002.

[32] S. Argamon, M. Koppel, J. W. Pennebaker, and J. Schler, "Automatically profiling the author of an anonymous text," *Communications of the ACM*, vol. 52, no. 2, pp. 119–123, 2009.

[33] E. Stamatatos, "Author identification: using text sampling to handle the class imbalance problem," *Information Processing & Management*, vol. 44, no. 2, pp. 790–799, 2008.

[34] C. E. Chaski, "Who' s at the keyboard? Authorship attribution in digital evidence investigations," *International Journal of Digital Evidence*, vol. 4, no. 1, pp. 1–13, 2005.

[35] F. J. Tweedie, S. Singh, and D. I. Holmes, "Neural network applications in stylometry: the Federalist Papers," *Computers and the Humanities*, vol. 30, no. 1, pp. 1–10, 1996.

[36] R. Zheng, J. Li, H. Chen, and Z. Huang, "A framework for authorship identification of online messages: writing-style features and classification techniques," *Journal of the American Society for Information Science and Technology*, vol. 57, no. 3, pp. 378–393, 2006.

[37] J. F. Burrows, "Word-patterns and story-shapes: the statistical analysis of narrative style," *Literary and Linguistic Computing*, vol. 2, no. 2, pp. 61–70, 1987.

[38] A. Jamak, A. Savatić, and M. Can, "Principal component analysis for authorship attribution," *Business Systems Research*, vol. 3, no. 2, pp. 49–56, 2012.

[39] D. I. Holmes, "A stylometric analysis of mormon scripture and related texts," *Journal of the Royal Statistical Society. Series A (Statistics in Society)*, vol. 155, no. 1, pp. 91–120, 1992.

[40] T. Mikolov, G. Corrado, K. Chen, and J. Dean, "Efficient estimation of word representations in vector space," in *Proceedings of International Conference on Machine Learning*, pp. 1–12, Atlanta, GA, USA, 2013.

[41] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proceedings of International Conference on Machine Learning*, pp. 1188–1196, Beijing, China, June 2014.

[42] P. Juola, "An overview of the traditional authorship attribution subtask," CLEF (Online work. Notes/Labs/Workshop), pp. 37–41, 2012, http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.636.449&rep=rep1&type=pdf.

[43] S. Bird and E. Loper, "NLTK: the natural language toolkit," in *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, pp. 1–4, Barcelona, Spain, July 2004.

[44] I. Markov, E. Stamatatos, and G. Sidorov, "Improving cross-topic authorship attribution: the role of pre-processing," in *Proceedings of 18th Computational Linguistics and Intelligent Text Processing*, Budapest, Hungary, April 2017.

[45] H. Ding, I. Takigawa, H. Mamitsuka, and S. Zhu, "Similarity-based machine learning methods for predicting drug-target interactions: a brief review," *Briefings in Bioinformatics*, vol. 15, no. 5, pp. 734–747, 2013.

[46] J. D. Burger, J. Henderson, G. Kim, and G. Zarrella, "Discriminating gender on twitter," *Association for Computational Linguistics*, vol. 146, pp. 1301–1309, 2011.

[47] M. Omar, B.-W. On, I. Lee, and G. S. Choi, "LDA Topics : representation and evaluation," *Journal of Information Science*, vol. 41, no. 5, pp. 1–14, 2015.

[48] R. Rehurek and P. Sojka, "Software framework for topic modelling with large corpora," in *Proceedings of the Workshop New Challenges for NLP Frameworks*, pp. 45–50, Valletta, Malta, May 2010.