Journal of Computer Science

Original Research Paper

Authorship Attribution of Short Historical Arabic Texts using Stylometric Features and a KNN Classifier with Limited Training Data

[1]Fatma Howedi, [2]Masnizah Mohd, [1]Zahra Aborawi Aborawi and [1]Salah A. Jowan

[1]*Asmarya Islamic University Zliten, Libya*
[2]*Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, 43600 Bangi Selangor, Malaysia*

Article history
Received: 03-07-2020
Revised: 02-10-2020
Accepted: 03-10-2020

Corresponding Author:
Masnizah Mohd
Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, 43600 Bangi Selangor, Malaysia
Email: masnizah.mohd@ukm.edu.my

Abstract: Authorship Attribution (AA) is a task that aims to recognize the authorship of unknown texts based on writing style. Out of the various approaches to solve the AA problem, Stylometry is a promising one. This paper explores the use of a K-Nearest Neighbor (KNN) classifier combined with stylometry features to perform AA. This study indicates the robustness of KNN in performing AA on short historical Arabic texts written by different authors. To classify the texts according to the author, KNN was trained with a set of stylometry features including rare words, count characters and 2-, 3- and 4-grams character levels. Various feature set sizes ranging from 34 to 2000 were tested in the experiment. The experiments were conducted on limited training data with datasets consisting of 3 short texts per the author’s book. This method proved to be at least as effective as Information Gain (IG) when selecting the most significant n-grams. Moreover, the KNN classifier achieved high accuracy results with the best classification accuracy of up to 90%, except for the 5-KK using the 4-gram character level. This work contributes towards utilizing KNN for identifying the distinctive stylometry feature for robust AA identification in short historical Arabic texts.

Keywords: Arabic, Authorship Attribution, Character Features, KNN, Lexical, Stylometry

Introduction

Authorship Attribution (AA) is the process of identifying the author of anonymous texts by providing some samples of texts of a few authors as a training set, assuming that the anonymous text is written by one of the authors of the known text samples (Shaker and Corne, 2010; Nirkhi *et al*., 2014). AA is a kind of Text Classification (TC) task. However, AA is different from TC because the writing style in AA is equally as important as the text content, but in TC, only the latter is important. Additionally, with different data sources such as articles and books, feature sets and classifiers may behave differently in AA (Bozkurt *et al*., 2007). Therefore, these differences make AA more challenging than TC.

In general, AA is useful for resolving issues such as plagiarism, detection and resolving historical questions regarding unclear or disputed authorship. Recently, practical applications for AA have grown in areas such as criminal law, civil copyright law and computer security for tracking authors of computer virus source codes. The vast majority of AA has been dedicated to identifying the author of long texts ranging from single passages to book chapters. Recently, more works have been focusing on short text. The present paper focused mainly on the issue of short-text, which refers to the amount of training data available per author. Stylistic choices are commonly accepted in texts written by an author but frequently occur less in short texts (Luyckx and Daelemans, 2011). Therefore, working with short texts constitutes a particular challenge and requires a robust and reliable representation of these texts. One of the fundamental sub-problems of AA is the extraction of the most suitable features to represent the writing style of each author. This problem is known as *stylometry*. This paper used the K-Nearest Neighbor (KNN) classifier to classify AA by extracting various character n-grams and lexical feature vectors of the writing style per author, as

Science Publications

the style of a text can be used as a distinctive feature to identify its writer (Takçı and Ekinci, 2012). Also, the writing style can be analyzed using factors within the same document, or by comparing two documents by the same author (Menai, 2012).

Literature Review

Only a few studies in the field of AA have focused explicitly on data size. This area has not been probed in much depth in many languages yet since most stylometry research tends to focus on long texts by authors or multiple short texts, where the longer the texts; the better the identification (Ouamour and Sayoud, 2012). Stamatatos (2009) stated that text samples should be long enough to ensure the text features sufficiently represent the author's style.

Nevertheless, there is no consensus on the minimum text sample required. Some studies investigated AA for short texts and found that using 2,500 words per sample would hardly provide a reliable result (Eder and Maciej, 2010). Also, AA accuracy deteriorates with reduced training data size (Luyckx and Daelemans, 2011; Al-Sarem and Emara, 2019). This paper proposes an overall investigation into AA that addresses 30 different short texts written by 10 ancient Arabic travelers who wrote several books describing their travels. A special Arabic dataset called (AAAT) was built. In AAAT, the number of words per author ranged between 1,289 and 1,785. Traditionally, the reliable minimum for an authorial set is considered to be 100,000 words per author (Ramnial *et al*., 2016). Nevertheless, (Knaap and Grootjen, 2007) used texts no longer than a sentence. Their experiment showed that, for 2 out of 5 classification tests, the text was correctly classified.

As for Arabic texts, a few studies in the field of AA can be found, namely (Abbasi and Chen, 2005a; 2005b). These studies tested the dataset of Arabic forum messages written by 20 authors with 20 messages per author. The principal conclusion of their experiments obtained the best accuracy (94%), but the overall performance was lesser than that of the English language. Despite the notable results they have, the dataset is quite large to extract enough features.

Ouamour and Sayoud (2012) also investigated Arabic text AA using a small data size. A variety of character ngrams features and word n-grams were used. Their results yielded the best accuracy (80%) using Support Vector Machine (SVM) classifier. In another work on the same dataset and the same sets of features, Ouamour and Sayoud (2018), examined authorship of short historical Arabic texts using the following classifiers: SVM, Linear Regression (LR), Multi-Layer Perceptron (MLP) and a new fusion called Vote Based Fusion (VBF). The results of their investigations indicated that the classifiers scored different accuracies and the VBF gave the highest accuracy (90%) among these classifiers.

Shaker and Corne (2010) studied the task of AA on Arabic text as well. They tested a set of Arabic function words using a Hybrid of Evolutionary Search and Linear analysis. At the phase of training and testing, the used texts were divided into 2 chunks: The first with 1000 word chunks, while the second with 2000 word chunks. The best performance obtained was 87.63% accuracy when 2000 word chunks were used. Partly, they showed that at least about 1,000 words chunks are necessary to obtain adequate characterization of function word usage for the Arabic authors. Moreover, they stated that the longer the text is, the higher the performance. The disadvantage of their method is that it depends just on function words to discover the authors and these function words were identified to reflect the semantic of English function words of previous researches.

On the other hand, other studies (Al-Ayyoub *et al*., 2017) showed that, even with short texts, performance of the classifiers depends mainly on the feature types rather than on the text size. They applied three well-known classifiers Naïve Bayes (NB), SVM and Bayes Networks using stylometric features and Bag Of Words (BOW) methods for AA of Arabic articles. They also concluded that stylometric features can generate more accurate results under most settings. The notable of their study that, they tested their method on large dataset consisted of 14,039 short articles.

The same findings, were reported by (Ouamour *et al*., 2016) in which the authors examined the performance of Manhattan distance, Stamatatos distance LR, MLP and SVM. Two types of features were investigated: Character n-grams features and words using Arabic dataset. The length of text varies from 100 to 3,000 words per document. The results were quite interesting, showing that the minimum textual size required to obtain a fair AA solution depends on both the feature types and classification methods.

Furthermore, in the same work of (Ouamour *et al*., 2016) reported that the optimal data size for a good AA is at least 2,500 words per sample. The results confirmed the findings by (Eder and Maciej, 2010) for the English language and the minimum data size of textual is 2,500 words per sample. Their results are useful, however we cannot extend them to every feature or classifier.

Recently, (Al-Sarem and Emara, 2019) investigated the effect of increasing training set size on the performance of attribution classifiers in the context of short religious Arabic texts. They used dataset consisted of 4,631 short texts. Mahalanobis distance, MLP and LR classifiers were employed. They stated that by increasing the size of training set, accuracy of the MLP classifier increased then decreased vastly. With some nuance change, the same thing was notated with Mahalanobis distance and LR classifiers. Interesting results could be notated in Al-Sarem and Emara (2019) where the n-gram features lead to decrease the performance of the classifiers with increasing the size of training set. However, generally speaking, the n-gram approaches provide the best results among the all stylometric features.

Regarding stylometric features, a study presented by (Nirkhi *et al*., 2014) investigated the use of stylometric features for AA in short texts of online English messages. For evaluating the performance of the classification methods such the SVM and KNN classifiers. The performance of SVM obtained 92% average accuracy and was higher than KNN (80% average accuracy). This study proved that stylometric features provides a way to classifiers that require fewer input variables than traditional statistics.

In summary, different classifiers were tested for AA in short texts. The SVM classifier is the most used classifier in the literature. The results of the literature showed that the classifiers have different behaviors regarding the used features and length texts. Thus we purpose to investigate the performance of KNN classifier on short historical Arabic texts ranging between 1289 and 1785 words per author. The length text varies from 290 to 800 words per document. Thus, we aim to train KNN classifier on limited data. For the purpose different stylometric features used. Additionally, methods of n-fold cross validation and Feature Selection (FS) were used for enhancing KNN Performance.

In the following sections, the AA approach used in this study is described.

Methodology

The common method used for solving the AA issue begins with a set of training data whose authors are known. Then, a set of features is extracted. These features are used in the ML algorithms for the classification process. This step allows the researcher to classify a test document whose author is unknown. Stylometric feature set and classification method used in this study are presented in next sections:

Stylometric Features

Stylometry is a behavioral feature that an author exhibits throughout his writing. Therefore, stylometry can be extracted and potentially used for checking the identity of the author of texts (Brocardo *et al*., 2013). Stylometry mainly relies on the assumption that individuals have distinctive ways of writing and this writing style cannot be manipulated consciously (Kusakci, 2012). Some examples of stylometric features include sentence length, word length, letter frequencies, word n-grams, character n-grams and function words. The basic categorization of these features is based on character, lexical, syntactic and semantic features (Oliveira Jr. *et al*., 2013). In this study, several characters and lexical features were tested. Table 1 represents the description of each purposed feature with an example. In the following, the main features employed in the proposed system are listed:

1. Character N-grams: These features provide information about the author’s style (or at least the topic of interest), which cannot be determined using only lexical features (Schwartz *et al*., 2013). Besides, character n-gram frequency is helpful to reliably handle limited data, which is why this parameter needs to be tested to facilitate short-text AA. A variety of character n-grams and words were used in another work (Ouamour and Sayoud, 2018), with the results yielding the best score of 90% accuracy. Meanwhile, Türkoğlu *et al.* (2007) focused on extracting bi-gram and tri-gram features using different classifiers including the KNN classifier. They concluded that n-grams yielded more successful results than additional features with allpurpose classifiers. The Character N-grams are strings of n consecutive characters from a given text (Stamatatos, 2009). Consequently, we distinguish character level n grams. For instance, for the text “the data” all character level 4-grams that can be generated are: "the_", "he_d", "e_da", "_dat", "data". Where the underscore character (_) represents the space, as is the convention in this study. Character such as “space” can provide vital information about the author’s style (Takçı and Ekinci, 2012). For the Arabic language, all character level 4-grams that can be generated from the text “ عدد الكلمات” are: "الكل", "لكلم", "كلما", "لمات عدد", "_", "_الك", "د_ال", "دد_ا" ". Noted that, we consider the Arabic texts from right to left
2. Character Count of Alphabets: According to these features, a text is viewed as a sequence of individual characters, so simple character level measures can be defined as a character count (Chen *et al*., 2012). For example, all generated features of this type of the text “data size” are: “d”, ”a” ,”t”, ”a”, “_”, ”s”, “i”, ”z”, ”e”. While, all generated features of this feature type from the Arabic text “ عدد الكلمات” are: "ع", "د", "د", "_", "ا", "ل", "ك", "ل"", "م", "ا", "ت".
3. Rare Words Frequency: Rare words are highinformation words since they have many lexical markers. They are the words that are repeated in a text at a low frequency. In this research, every single word that appeared once or twice in each document per author was considered a rare word feature

Table 1: Description of each stylometric features with an example

Stylometric Features name	Description	Example for the text (the table)
Character Count of Alphabets	Individual characters	t, h, e, _, t, a, b, l, e
Bi-gram Character (2-gram)	Two consecutive characters	th, he, e_, _t, ta, ab, bl, le
Tri-gram Character (3-gram)	Three consecutive characters	the, he_, e_t, _ta, tab, abl
Tetra-gram Character (4-gram)	Four consecutive characters	the_, he_t, e_ta, _tab, tabl
Rare Words	Low frequency	Single word appeared once or twice in the text

KNN Classifier

The K-Nearest Neighbor (*KNN)* algorithm is amongst the simplest Machine Learning (ML) algorithms. It is a type of instance-based learning, which runs local approximations. All computations are deferred until classification. An object is classified by a majority vote of its neighbors with the object being assigned to the class most common amongst its *k* nearest neighbors (Nirkhi *et al*., 2014). Here, *k* means a small positive integer. If *k* = 1, then, the object is simply assigned to the class of that single nearest neighbor. Ramnial *et al*., (2016; Nirkhi *et al*., 2014) applied two ML algorithms, KNN and Sequential Minimal Optimization (SMO), using stylometric features. All results yielded an accuracy of 90%, except for the KNN classifier. The KNN classifier is chosen as a classification method for the following reasons:

- It is simple but powerful classification. KNN algorithm has ability to distinguish new instance with limited training data available, since it only does more work during classification and then it can obtains prediction with good probability and relatively inexpensive computational resources as well. Thus, the KNN can outperform learningbased classification method when the amount of learning data is limited
- To the best of our knowledge, the KNN classifier has not been considered before, for the problem of AA with limited training data of short Arabic texts. The SVM classifier was the most used classifier with variety features in many studies of AA in Arabic and other language

In this study, the steps for the AA task included preprocessing, feature extraction, classification and author identification. A flowchart illustrating the text processing and classification process in this research is shown in Fig. 1.

Data Pre-Processing

Data pre-processing is a crucial step in AA. Text documents in their original form are not suitable for learning. These documents must thus be converted into a vector space since most learning algorithms use attribute-value representation (Elayidom *et al*., 2013; Abu-Hamad and Mohd, 2019; Salam and Kadir, 2017). In this study, the AA dataset was sent to a preprocessing algorithm, which was built (using C# language) based on the following steps:

1. Tokenization: tokenization is a method of splitting a stream of text input into meaningful elements (Elayidom *et al*., 2013). These elements are called tokens, for example, symbols, words, phrases and so on. The extracted group of tokens serves as input for further processes such as parsing and text mining, which, in turn, are part of lexical analysis (Elayidom *et al*., 2013). In this study, the dataset was processed into grams of 2, 3 and 4 character grams, by tokenizing the characters on white space. White spaces were considered as character and they replaced with underscore character (_). Table 2 shows character grams and character count of the Arabic text " **وكان ارتحالي**" as a sample from the AAAT dataset after applying the tokenization process
2. Punctuation Mark Removal: All the punctuation marks (e.g., "!\:;?.,،) were removed from the texts of each document
3. Normalization: Is the process of finding the standard form of all letters found in each document in a dataset (Al-Badarenah *et al*., 2016). Normalization is used to help overcome the variation in text representation (Altheneyan and Menai, 2014; Omar *et al*., 2013; Saad and Latiff, 2018). In this study, some Arabic letters such as (alef) were normalized into all their forms such as (آ, إ, أ) to (ا). Also, the final ةwas replaced with ه and the final ىwas replaced with ي. All these letters were converted to the same case of their forms to more accurately reflect the dimensionality of the vector space. Also, numbers such as the author's dates were cleaned. This step was done because this type of information may cause an unfair advantage on the controlled dataset that will not scale the authors into genres or topics (Luyckx, 2010)

This text data pre-processing step is crucial for determining the quality of the text stages and includes the feature extraction and classification stages.

Extracting Features

The features and their extraction are dependent on the text language. The features are extracted from the authors' text and can be used to understand the peculiarity of an author's writing. Different character and lexical features are extracted here, including rare words, character count, character bi-gram, character trigram and character tetra-gram.

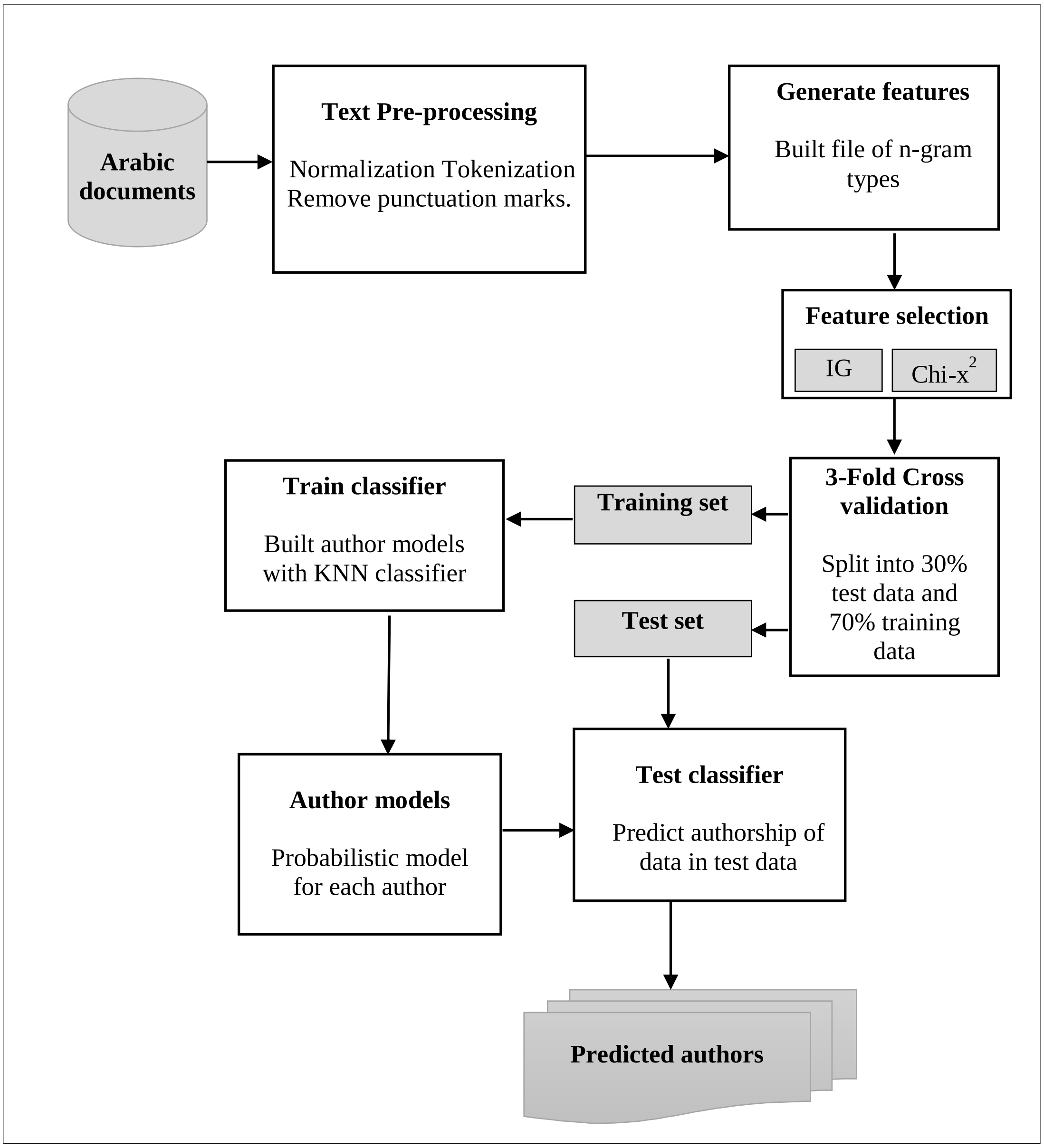
Arabic documents
Text Pre-processing
Normalization Tokenization Remove punctuation marks.
Generate features
Built file of n-gram types
Feature selection
IG
Chi-x²
3-Fold Cross validation
Split into 30% test data and 70% training data
Training set
Test set
Train classifier
Built author models with KNN classifier
Author models
Probabilistic model for each author
Test classifier
Predict authorship of data in test data
Predicted authors

Fig. 1: Flowchart of text processing and classification process in AA

Table 2: Generated features of the Arabic text “ **وكان ارتحالي**” after tokenization process

Original text from the AAAT dataset	وكان ارتحالي
Character count	و, ك, ا, ن, _, ا, ر, ت, ح, ا, ل, ي
The 2-grams that were generated	وك, كا, ان, ن_, _ا, ار, رت, تح, حا, ال, لي
The 3-grams that were generated	وكا, كان, ان_, ن_ا, _ار, ارت, رتح, تحا, حال, الي
The 4-grams that were generated	وكان, كان_, ان_ا, ن_ار, _ارت, ارتح, رتحا, تحال, حالي
Rare Words	ارتحالي

Feature Selection and Reduction Methods

Some features such as character and lexical features can considerably increase the dimensionality of the feature set (Stamatatos, 2009; Howedi and Mohd, 2014). In such cases, Feature Selection (FS) methods such as information gain can be used to reduce such dimensionality. Dimensionality can be reduced by selecting just a subset of the original features. Some features can be removed based on the frequencies of those features, by setting those frequencies greater than or less than a defined threshold value (Al-Badarenah *et al.*, 2016). Many data mining algorithms perform better with lower dimensionality because the most characteristic features will remain after FS (Fissette, 2010). As reported here, Information Gain (IG) and Chisquared (*Chi-*x^2) were used as the FS technique:

(1) *Chi-Squared (Chi-*x^2*):* Is a statistical method that measures divergence from the expected distribution, assuming that feature occurrence is independent of class value
(2) *Information Gain (IG):* Considers each feature independent of others and offers a ranking of the features based on their IG score so a certain number of features can be selected easily

Classification Model

As is the case of this study, if the researcher only has a small dataset to work with for the classification problem, it would be difficult to provide enough data for separate training sets and testing sets. In this case, it is possible to apply the n-fold cross-validation technique, by using all the data as both training and testing data. Thus, the cross-validation technique was applied to provide a more meaningful result (Taş *et al.*, 2007; Burrows, 2010). Cross validation also helps to avoid over-fitting and provides an unbiased estimate of the learning algorithm predictive performance (Keevers, 2019). By dividing the dataset randomly into *n* partitions, called folds. One of the n partitions is keep as the testing data and the rest of the *n*-1 partitions are used as training data, the training and testing data must not have any of the same data points. Then the classifier will be trained n times. For each training run a single part from n partition, will be selected as the test set and using the rest for training. Then fitting a model on the training set and evaluated it on the test set. The model will be discarded after storing the evaluation score.

In this study, we set *n* = 3 that means the AAAT dataset was divided into 3 partitions (folds), each fold has 10 text files from the AAAT dataset. Then three models were trained and evaluated with each fold given a chance to be the held out test set, where each model trained on 2 unique training folds. Each model also tested on a unique test fold. Therefore, the classification was performed thrice:

- Model1: Trained on Fold1+ Fold2, Tested on fold3
- Model2: Trained on Fold2+ Fold3, Tested on fold1
- Model3: Trained on Fold3+ Fold1, Tested on fold2

To obtain a final accuracy measure, each classification model was then discarded after retaining the evaluation score. The skill scores then summarized to use.

Evaluation

Macro averaged precision ($\mathrm{Pr}^{(M)}$) and recall ($\mathrm{Re}^{(M)}$) were used to evaluate this work. Furthermore, macro-averaged F_1-measure $\left(F_1^{(M)}\right)$ was also used to compare the experiments so that increases or decreases in classification efficiency could be measured. The resulting scores were evaluated by computing the number of True Positives (TP) and True Negatives (TN) over all the experiments and calculating the precision ($\mathrm{Pr}^{(M)}$), recall ($\mathrm{Re}^{(M)}$) and F_1-measure $\left(F_1^{(M)}\right)$ per Equations (1) to (3) below:

$$\mathrm{Pr}^{(M)} = \frac{\sum_{i=1}^{n} Precision,_{\textit{for each authorship class}}}{Total\ number\ of\ authorship\ classes} \tag{1}$$

$$\mathrm{Re}^{(M)} = \frac{\sum_{i=1}^{n} Recall,_{\textit{for each authorship class}}}{Total\ number\ of\ authorship\ classes} \tag{2}$$

$$F_1^{(M)} = \frac{\sum_{i=1}^{n} F_1\ ,_{\textit{for each authorship class}}}{Total\ number\ of\ authorship\ classes} \tag{3}$$

Experiment and Results Discussion

A set of experiments was run to evaluate the effect of short Arabic texts with limited training data (two short text documents per author) on different features to show the robustness of the KNN performance. Moreover, the effect of feature size was tested via IG and Chi-x^2 FS methods.

Dataset Description

The study considers a standard dataset of short texts as an approximation of the ancient Arabic texts. Ten different authors wrote ten books. One book per author was chosen from "Alwaraq library" website, as in the AAAT dataset (i.e., Authorship attribution of Ancient Arabic Texts). Additionally, three pages were selected from each book, each to be stored as one page in a text file. According to Table 3, the average length of each file was 550 words. This allows probing into the scalability of the approach with limited training data and short texts documents.

Overall Results

As can be seen from Table 4, a good attribution score of 90.42% average accuracy was obtained by applying 5-NN with features of tetra-gram characters. This score was the best score of all the features employed in the separately-carried-out experiments. Additionally, a good average accuracy of 89.29 and 88.33% were obtained with the features of rare words using 5-NN and 3-NN, respectively. Moreover, character-based ngrams are better than character counts, which only scored 23.33% of best attribution.

Also, it can be observed from Table 4 that the average good attribution score was an accuracy of 62.83% with 5-NN compared to 3-NN, which achieved a score of 61.84%. This result shows that the KNN model is more stable when constructed with more neighbors. However, there is no direct relationship between predication performance and range of neighborhood. That because, based on the result of experiments in Table 7 and 8, features size also has impact on prediction performance, since it can be noted that each value of K (3-NN and 5-NN) produced different accuracies depends on features size. For instance, when Rare words feature and IG were used with KNN (Table 7): The prediction of KNN when the number of neighbors = 5 produced different average precisions (83.17, 89.29 and 49.91%) depending on features size of (100, 500, 1000) respectively. Also, depending on features size of (100, 500, 1000) the prediction of KNN when the number of neighbors = 3 produced different average precisions of (88.33, 78.33 and 23.75%) respectively, so that, each prediction of KNN has different performance depends on both the number of neighbors (k) and the number of features size available. This proves that the size of features and the range of neighborhood both have an impact on the prediction performance of KNN.

It is important to mention that the average accuracy of 90.42, 89.29 and 85.00% with limited training samples is relatively high, where several previous works by (Ramnial *et al*., 2016) stated that, with 10,000 words per author, the average accuracy is high and reduced with 1000 words. Also, (Eder and Maciej, 2010) stated that text should not be less than 2500 words per sample to obtain good results. On the other hand, this paper presented short texts ranging between 1289 and 1785 words per author.

Feature Selection for Enhancing KNN Algorithm Performance

The aim of Feature Selection (FS) methods is to eliminate the useless feature. To maximize the success of the Authorship identification system and reduce the size of the dimensionality of the vector space. Bay and Çelebi (2016). The effect of FS methods on KNN performance was also tested. The FS methods applied on 3-KK and 5-KK of the KNN value using the different features. We used IG and Chi-x^2 in RapidMiner tool. We conducted the experiments in two different ways. In the first way, we applied the 3- NN and 5-NN separately to each feature condition before applying IG and Chi-x^2. In the second way, different experiments were conducted, by reducing our features set to different sizes by eliminating the worst features based on IG and Chi-x^2 processes, then we applied the 3-NN and 5-NN separately to each feature condition with different sizes. Improvements in the KNN performance rates after applying FS procedure can be observed in Table 5.

Table 3: Size of texts in terms of words

Author Designation	Author name (En)	Text file (1)	Text file (2)	Text file (3)	Average text length	Total No. words
Author1	Ibn Batuta	630	605	308	514	1543
Author2	Ibn Jubayr	575	540	598	571	1713
Author3	Nasser Khasru	657	**800**	**290**	582	1747
Author4	Ibn Fathlan	599	593	593	595	**1785**
Author5	AlMuja-zwer	459	511	722	564	1692
Author6	Al Yussee	511	559	636	568	1706
Author7	Lessan Addin	599	460	541	533	1600
Author8	Al Alussi.	515	653	578	582	1746
Author9	Al Hamawi	322	629	548	499	1499
Author 10	Al Balwi	591	345	353	429	**1289**

Table 4: Percentage accuracy of good attribution obtained using KNN with different features and features sizes

	Accuracy of good attribution using the KNN classifier	
Feature type	3-NN	5-NN
Character count	9.21%	11.40%
Char. Bi-gram	43.33%	50.00%
Char. Tri-gram	85.00%	71.50%
Char. Tetra-gram	83.33%	**90.42%**
Rare words	**88.33%**	**89.29%**
Avg. of good attributions	61.84%	62.83%
Best score	90.42%	

Table 5: Accuracy of good attribution in % before and after Feature Selection (FS)

	Accuracy % using 3-NN		Accuracy % using 5-NN	
Feature	Before FS	After FS	Before FS	After FS
Character count	20.00	23.33	23.33	23.33
Char. Bi-gram	36.67	43.33	23.33	50.00
Char. Tri-gram	60.00	85.00	60.00	71.50
Char. Tetra-gram	60.00	83.33	53.33	**90.42**
Rare words	56.67	88.33	66.67	89.29
Best score	60.00	88.33	**66.67**	**90.42**

Table 6: Generated features and best feature size selected for each feature type

Feature type	Total number of generated features	Top-k of selected feature size
Character count	34	30
Char. Bi-gram	803	100, 300, 500.
Char. Tri-gram	7412	100, 500, 1000.
Char. Tetra-gram	23524	100, 500, 1000, 2000.
Rare words	7547	100, 500, 1000

Table 7: Results of average precision in % using KNN with different feature selection sizes via IG and *Chi-x*^{2} methods

		KNN-3		KNN-5	
Type of feature	The selected feature size	*Chi x*^{2}	IG	*Chi x*^{2}	IG
Char. Count	30	9.21	11.21	9.40	**11.40**
2-gram **(Bi)**	100	28.56	**43.94**	30.83	46.31
	300	32.14	38.36	30.70	37.02
	500	21.05	36.30	17.62	17.53
3-gram **(tri)**	100	51.36	35.67	33.75	33.48
	500	39.91	50.83	33.88	48.89
	1000	22.26	**85.00**	45.48	71.50
4-gram **(**Tetra**)**	100	53.75	72.67	40.67	58.67
	500	75.17	73.17	50.00	**90.42**
	2000	61.06	83.33	49.17	82.25
Rare words	100	20.00	88.33	9.83	83.17
	500	66.61	78.33	73.56	**89.29**
	1000	57.50	23.75	46.33	49.91

Table 8: Results of Average Recall (in %) using KNN with different feature selection sizes via IG and *Chi-x*^{2} methods

		KNN-3		KNN-5	
Type of feature	Selected feature size	*Chi x*^{2}	IG	*Chi x*^{2}	IG
Character count	30	23.33	20.00	23.33	**23.33**
2-gram **(**Bi**)**	100	36.67	43.33	33.33	**50.00**
	300	33.33	40.00	23.33	43.33
	500	26.67	40.00	20.00	23.33
3-gram **(**Tri**)**	100	53.33	46.67	36.67	40.00
	500	33.33	56.67	36.67	53.33
	1000	30.00	**76.67**	43.33	73.33
4-gram **(**Tetra**)**	100	53.33	73.33	40.00	56.67
	500	70.00	76.67	53.33	**80.00**
	2000	40.00	70.00	43.33	70.00
Rare words	100	30.00	**83.33**	20.00	76.67
	500	66.67	70.00	63.33	76.67
	1000	53.33	23.33	43.33	36.67

The Effect of Short-Text Documents on Feature Selection using Different Feature Sizes

Table 6 summarizes the total number of each generated feature condition obtained from the AAAT dataset. Also, it shows the top-k feature size that was selected from the total number of each generated feature (e.g., the top-k frequent features when k = 500 means the most 500 frequent features were selected). Each top-k feature size that contains the best features was fed separately into the KNN classifier with the highest information gain or *Chi-x*^{2} value.

According to Table 7, the results of average precision indicate that with the most used features the IG achieved better results (between 11.21 and 90.42%) than the Chi-*x*^{2} (between 9.21 and 75.17%), with both cases applying 3-NN and 5-NN of the KNN value; applied separately to each feature condition. Besides, the best results obtained from both the IG and Chi-*x*^{2} were obtained using 5-NN applied on rare words and Tetra-gram features with a feature size of 500.

In another experiment done separately for each feature using KNN; Table 8 shows the results of the average recall using different feature sizes weighted with IG and then Chi-*x*^{2}.

Table 8 shows that the best average recall (83.33%) obtained for both IG and Chi-*x*^{2} was recorded using 3-KK with the rare words feature and with a feature size equal to 100 IG. Likewise, the IG achieved better results than the Chi-*x*^{2} with the most features, where, in both cases, the 3-NN and 5-NN of the KNN classifier were applied. From the results, the most suitable feature set could be obtained according to the outcomes of Re, Pr and F$^{\cdot}_{1}$, per Fig. 2.

To summarize, it can be noted that different values of attribution results were obtained by applying KNN with IG and Chi-x^2 methods using different feature sizes of 100, 300, 500, 1000 and 2000. This result proves that the size of the features has an impact on the performance of the attribution. This is because feature size has an impact on frequency, which is considered the most important criterion for feature selection. In general, the more frequent the features; the more stylistic the variation that it captures (Putniņš *et al*., 2006).

Furthermore, Chi-x^2 did not work on short texts, as it yielded worse results than IG. The reason behind that was supported by the data, where the texts used in this study were very short to allow the regular reoccurrence of characters. The Chi-x^2 begins to perform better on larger data which can produce higher dimensional feature space. However, character n-grams features can considerably increase the dimensionality, the texts size was not enough to produce higher dimensionality. According to the results presented in Tables 7 and 8, indicate that the Chi-x^2 achieved best results (obtained to 75.17%) by using Tetra-gram feature, which produced higher dimensionality than character count and Bi-gram which achieved results 23.33 and 50.00% of good attribution respectively. Previous work done by (Mohsen *et al*., 2016) showed that Chi-*x*^{2} can outperforms other FS method only when high dimensional feature space are used. Nicolosi (2008) stated that Chi-x^2 is more suitable on larger data. On the other hand, the present investigation demonstrated that the IG can work well on low dimensionality feature space and outperforms Chi-*x*^{2}.

Nevertheless, the Chi-*x*^{2} was faster in its implementation of the IG algorithm, which took more time, which is more than 40 min in the experiments run.

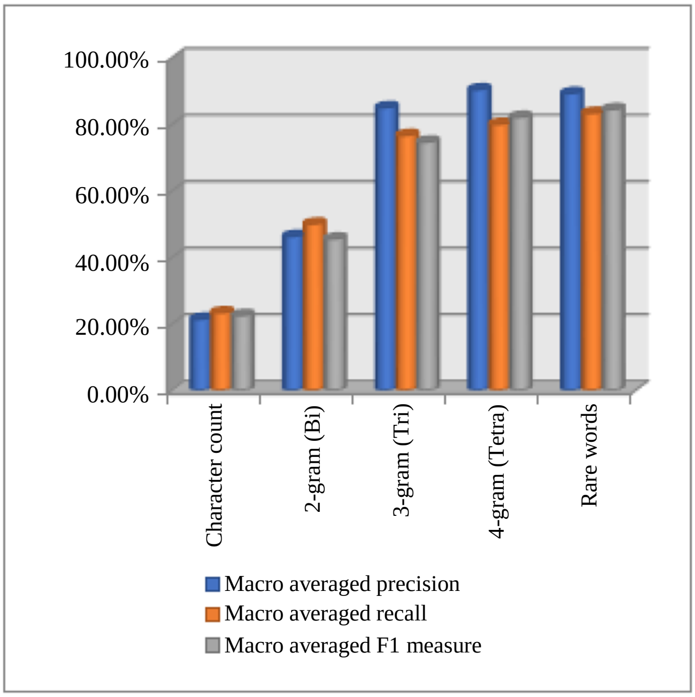

Fig. 2: Results obtained for each feature set according to precision (Pr$^{(M)}$), recall (Re$^{(M)}$) and F_1-measure (F_1)

Table 9: The best average Pr$^{(M)}$ and Re$^{(M)}$ according to total number of each feature

Feature type	Total number of generated features	The best average precision	The best average recall
Character count	34	11.40%	23.33%
Char. Bi-gram	803	43.94 %	50.00%
Char. Tri-gram	7412	85.00%	76.67%
Char. Tetra-gram	23524	90.42%	80.00%
Rare words	7547	89.29%	83.33%

Lastly, a slight failure was also noted using bi-gram features (attribution accuracy scores of about 40% and 50% were obtained). This is because the regular reoccurrence of bi-gram characters was low in such short texts. Also, the character count feature failed, it obtained a very low classification result (11.40%) for average precision. The reason behind that is the generated features size of this type of features was very small, only 34 features. In such cases, the character count feature may not have enough information to make the best decision. So that, we can observe that fewer feature items have lower accuracy as shown in Table 9. There is a large different in Pr$^{(M)}$ and Re$^{(M)}$ between character count feature and other used features.

Comparison with Related Works

In this section we consider the recent and closer works to our work since most of previous related studies investigated the problem of AA in multi-short Arabic texts. Study such as Al-Ayyoub *et al.* (2017) which considered dataset consisted of 14,039 short articles written by 42 authors and Al-sarem and Emara, (2019) used large dataset consisted of 4,631 short text documents distributed among 15 authors. On the other hand, we investigated the performance of KNN with small dataset consisted only 30 short Arabic texts written by 10 authors as the case study of (Ouamour and Sayoud, 2018; 2012). So that, we decided to compare our approach to (Ouamour and Sayoud, 2018; 2012) works. Siham and Halim (2018) applied three classifiers: LR, MLP SVM and Vote Based Fusion technique with different features set of character and word n-grams. Siham and Halim (2012) used the same features set with SVM. We discovered that our approach using KNN, which was enhanced by FS methods has achieved the best accuracy (90.42%) while the second best accuracy (90.00%) was obtained by (Siham and Halim, 2018) using Vote Based Fusion method with MLP, followed by LR achieved 70% accuracy and SVM obtained 80% accuracy. The comparison gives an indication of different classifiers performance.

Conclusion

This paper presented *a new AA task to investigate* the use of a KNN classifier for the Authorship Attribution (AA) of short Arabic texts to determine the robustness of this method under different lengths (varies from 290 to 800 words) of text samples used for the training. The KNN was trained on limited data against two text documents per author, where the average text length was about 550 words per document. Several state-of-the-art features were tested for the Arabic language, with experiments carried out separately for each feature condition, including rare words, count characters and character level (Bi-gram, Tri-gram and Tetra-gram). The last set of tests evaluated the effects of feature size using various feature set sizes by applying the IG and Chi-x^2 selection methods. Some noteworthy points of this a new AA task are listed below:

- Although the size of the texts used in this study was small (ranging between 1289 and 1785 words per author) the performance of the KNN classifier was interesting (90.42% average accuracy for the best score)
- The character tetra-gram and rare words features have the best performance. These feature sets are very effective even with limited training data size. On the other hand, classification failure was observed when the character count feature was used
- Feature selection methods are necessary to achieve an outstanding performance of classifiers
- Information Gain (IG) selection method is more suitable with short texts than Chi-x^2
- Our results show that using about 2000 words per author, the authors of Arabic short texts can be successfully identified
- This work on AA is one of the few works done on short Arabic texts, so it serves as real motivation to conduct more AA investigation on the Arabic language

In the future, a new set of stylometric features could be used to enhance the performance of the KNN classifier.

Acknowledgment

This study was supported by the Universiti Kebangsaan Malaysia grant DCP-2017-007/4 and PPFTSM- 2020.

Author’s Contributions

Fatma Howedi: Researching and conducting the experiment as well as writing the manuscript.

Masnizah Mohd: Provide publication recommendations, reviewing the manuscript as well as supporting the publication of this manuscript.

Zahra Aborawi Aborawi and Salah A. Jowan: Reviewing the manuscript and provided guidance during project experimentation phase.

Ethics

This is an original manuscript and contains unpublished material. All authors have read, reviewed and approved the manuscript and there are no ethical issues involved.

References

Abbasi, A., & Chen, H. (2005a, May). Applying authorship analysis to Arabic web content. In International Conference on Intelligence and Security Informatics (pp. 183-197). Springer, Berlin, Heidelberg.

Abbasi, A., & Chen, H. (2005b). Applying authorship analysis to extremist-group web forum messages. IEEE Intelligent Systems, 20(5), 67-75.

Abu-Hamad, M., & Mohd, M. (2019). Data Categorization and Model Weighting Approach for Language Model Adaptation in Statistical Machine Translation. International Journal of Advanced Computer Science And Applications, 10(1), 135-141.

Al-Ayyoub, M., Alwajeeh, A., & Hmeidi, I. (2017). An extensive study of authorship authentication of arabic articles. International Journal of Web Information Systems.

Al-Badarenah, A., Al-Shawakfa, E., Al-Rababah, K., Shatnawi, S., & Bani-Ismail, B. (2016). Classifying Arabic text using KNN classifier. International journal of advanced computer science and applications, 7(6).

Al-Sarem, M., & Emara, A. H. (2019). The effect of training set size in authorship attribution: application on short Arabic texts. International Journal of Electrical & Computer Engineering (2088-8708), 9(1).

Altheneyan, A. S., & Menai, M. E. B. (2014). Naïve Bayes classifiers for authorship attribution of Arabic texts. Journal of King Saud University-Computer and Information Sciences, 26(4), 473-484.

Bay, Y., & Çelebi, E. (2016). Feature selection for enhanced author identification of Turkish text. In Information Sciences and Systems 2015 (pp. 371-379). Springer, Cham.

Bozkurt, I. N., Bağlıoğlu, Ö., & Uyar, E. (2007). Authorship attribution: performance of various features and classification methods. In 22nd International Symposium on Computer and Information Sciences, ISCIS 2007-Proceedings (pp. 158-162). IEEE.

Brocardo, M. L., Traore, I., Saad, S., & Woungang, I. (2013, May). Authorship verification for short messages using stylometry. In 2013 International Conference on Computer, Information and Telecommunication Systems (CITS) (pp. 1-6). IEEE.

Burrows, S. (2010). Source code authorship attribution.

Chen, Z., Huang, L., Yang, W., Meng, P., & Miao, H. (2012). More than word frequencies: Authorship attribution via natural frequency zoned word distribution analysis. arXiv preprint arXiv:1208.3001.

Eder & Maciej. (2010, July). Does Size Matter? Authorship Attribution, Small Samples, Big Problem. In DH (pp. 132-134).

Elayidom, M. S., Jose, C., Puthussery, A., & Sasi, N. K. (2013). Text classification for authorship attribution analysis. arXiv preprint arXiv:1310.4909.

Fissette, M. (2010). Author identification in short texts. Bachelor's Thesis, Department of Artificial Intelligence, Radboud University.

Howedi, F., & Mohd, M. (2014). Text classification for authorship attribution using Naive Bayes classifier with limited training data. Computer Engineering and Intelligent Systems, 5(4), 48-56.

Keevers, T. L. (2019). Cross-validation is insufficient for model validation. Joint and Operations Analysis Division, Defence Science and Technology Group: Victoria, Australia.

Knaap, L., & Grootjen, F. A. (2007). Author identification in chatlogs using formal concept analysis.

Kusakci, A. O. (2012, September). Authorship attribution using committee machines with k-nearest neighbors rated voting. In 11th symposium on neural network applications in electrical engineering (pp. 161-166). IEEE.

Luyckx, K. (2010). Scalability Issues in Authorship Attribution. PhD thesis, Department of Linguistics, Faculty of Arts and Philosophy, Dutch UPA University.

Luyckx, K., & Daelemans, W. (2011). The effect of author set size and data size in authorship attribution. Literary and linguistic Computing, 26(1), 35-55.

Menai, M. E. B. (2012). Detection of plagiarism in Arabic documents. International Journal of Information Technology and Computer Science, 10(10), 80-89.

Mohsen, A. M., El-Makky, N. M., & Ghanem, N. (2016, December). Author identification using deep learning. In 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA) (pp. 898-903). IEEE.

Omar, N., Mohd, M., & Jamat, Y. (2013). Automatic Probability in the Bayes Algorithm for Conversational Agent Application. Asia-Pacific Journal of Information Technology and Multimedia. 2(1), 27-37.

Nicolosi, N. (2008). Feature selection methods for text classification. Department of Computer Science, Rochester Institute of Technology, Tech. Rep.

Nirkhi, S. M., Dharaskar, R. V., & Thakare, V. M. (2014). Authorship Attribution of online messages using Stylometry: An Exploratory Study. In International Conference on Advances in Engineering and Technology (ICAET'2014).

Oliveira Jr., W., Justino, E., & Oliveira, L. S. (2013). Comparing compression models for authorship attribution. Forensic science international, 228(1-3), 100-104.

Ouamour, S., & Sayoud, H. (2012, June). Authorship attribution of ancient texts written by ten arabic travelers using a smo-svm classifier. In 2012 International Conference on Communications and Information Technology (ICCIT) (pp. 44-47). IEEE.

Ouamour, S., & Sayoud, H. (2018, September). A Comparative Survey of Authorship Attribution on Short Arabic Texts. In International Conference on Speech and Computer (pp. 479-489). Springer, Cham.

Ouamour, S., Khennouf, S., Bourib, S., Hadjadj, H., & Sayoud, H. (2016). Effect of the text size on stylometry—application on Arabic religious texts. In Advanced Computational Methods for Knowledge Engineering (pp. 215-228). Springer, Cham.

Putniņš, T., Signoriello, D. J., Jain, S., Berryman, M. J., & Abbott, D. (2006, January). Advanced text authorship detection methods and their application to biblical texts. In Complex Systems (Vol. 6039, p. 60390J). International Society for Optics and Photonics.

Ramnial, H., Panchoo, S., & Pudaruth, S. (2016). Authorship attribution using stylometry and machine learning techniques. In Intelligent Systems Technologies and Applications (pp. 113-125). Springer, Cham.

Saad, S., & Latiff, U. K. (2018). Extraction of concept and concept relation for islamic term using syntactic pattern approach.

Salam, Z. A. A., & Kadir, R. A. (2017). A Study of context influences in Arabic-English language translation technologies.

Schwartz, R., Tsur, O., Rappoport, A., & Koppel, M. (2013, October). Authorship attribution of micro-messages. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (pp. 1880-1891).

Shaker, K., & Corne, D. (2010, September). Authorship attribution in arabic using a hybrid of evolutionary search and linear discriminant analysis. In 2010 UK Workshop on Computational Intelligence (UKCI) (pp. 1-6). IEEE.

Stamatatos, E. (2009). A survey of modern authorship attribution methods. Journal of the American Society for information Science and Technology, 60(3), 538-556.

Takçı, H., & Ekinci, E. (2012). Character Level Authorship Attribution for Turkish Text Documents. Turkish Online Journal of Science & Technology, 2(3).

Taş, T., Görür, A. K., & Tufan, T. A. Ş. (2007). Author identification for Turkish texts. Cankaya University Journal of Arts and Sciences, 1(7), 151-161.

Türkoğlu, F., Diri, B., & Amasyalı, M. F. (2007, August). Author attribution of Turkish texts by feature mining. In International Conference on Intelligent Computing (pp. 1086-1093). Springer, Berlin, Heidelberg.