Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 117 (2017) 145–152

Procedia Computer Science

www.elsevier.com/locate/procedia

3rd International Conference on Arabic Computational Linguistics, ACLing 2017, 5–6 November 2017, Dubai, United Arab Emirates

Dynamic Similarity Threshold in Authorship Verification: Evidence from Classical Arabic

Hossam Ahmed[a]

[a]*Leiden University, Witte Singel 25, 2311 BZ Leiden, The Netherlands*

Abstract

Many Authorship Verification Machine Learning-based algorithms rely on establishing a similarity threshold θ between a candidate text and known texts in terms of one or more linguistic features. Documents that score below that threshold are rejected as not written by the same author. Current definitions of θ rely on both negative and positive training input. An algorithm that relies exclusively on positive training data, and dynamically calculates θ for each verification problem performs with good accuracy, tested using a training and evaluation corpus from Classical Arabic.

Peer-review under responsibility of the scientific committee of the 3rd International Conference on Arabic Computational Linguistics.

Keywords: Authorship Verification ; Classical Arabic ; Similarity Threshold

1. Introduction

Authorship Verification (AV) is a special type of Authorship Attribution (AA) problems. An AA problem is a classification problem where the author of a document D is selected from a set of candidate authors. Most current algorithms use a straightforward metric: collect known documents created by each of the candidate authors (S), isolate a linguistic feature or set of features in D and S, the author whose texts are most similar to D is the winner.

One major obstacle to implementing Machine Learning (ML) algorithms to AV, compared to AA, is the absence of negative evidence (documents that are not written by the author). Typically, we only have a candidate document, and a corpus of authentic work by the author. Most learners rely on both negative and positive evidence to calculate the probability that a candidate document is not written by the author. Attempts at solving this problem supplement the training corpus with a corpus of negative documents, converting the problem into an AA problem, or use

* Hossam Ahmed. Tel.: +31-(0)71-527-4417.
E-mail address: h.i.a.a.ahmed@hum.leidenuniv.nl

1877-0509
Peer-review under responsibility of the scientific committee of the 3rd International Conference on Arabic Computational Linguistics.
10.1016/j.procs.2017.10.103

supplemental negative training data in other ways, as will be discussed in the next section.

This approach, although helpful in simplifying the classification problem, does not guarantee optimal identification. The accuracy of the classifier is affected to a certain extent by the selection of the distractors (the negative training data), and not only by the candidate document and author corpus. If the negative data is too different from the texts (candidate or positive corpora), the classifier can over-generate with potentially reduced accuracy. The experimenter cannot intentionally analyze the distractors before using them to ensure that they are 'close enough,' as this will risk bias in the experiment design.

In this paper, I examine the effectiveness of an algorithm whereby only positive data is used for training. The context for this exploratory work is classical Arabic documents. [13] indicates that it is not unusual for authors with unorthodox ideas in the Middle Ages to attribute their work to a more reputable writer to improves its chances of circulation. Others would insert 'heretic' sections in their rivals' work to discredit them.

Using texts known to be authentic or forged works attributed to Al-Ghazali (1058 - 1111) as training and testing data, I examine the effectiveness of using token similarity in authorship verification. Specifically, positive data is used to establish a lexical similarity interval for authentic works. The dispersion of individual similarity values determines a confidence interval that delimit a similarity threshold within which test documents are deemed authentic. The availability of positive evidence only is common in Classical Arabic studies.

2. Related Work

There has been interesting work on AA in Arabic. [10] examines the efficiency of various ML classifiers to attribute short historical Arabic texts. It indicates that for short texts, statistical ML classifiers (SMO-SVM, and MLP) give better results than purely statistical (linear regression) and distance-based classifiers. They also indicate that the use of rare [1] words and individual words give better results than word n-grams, the former giving best results. [7] use the same dataset to examine the effectiveness of Naïve Bayes ML methods. [2] also examines Naïve Bayes methods in AA of classical Arabic texts, but using larger texts for their corpora (approx. 2000-word chunks). [11] examines the usefulness of function words in AA in modern Arabic books using Linear Discriminant Analysis (LDA). [1] uses linguistic and non-linguistic features (such as fonts and elongation) to analyze authorship of extremist-group web forum messages. [5] uses punctuation, function words and clitics in a variety of modern Arabic texts to show that it is possible to achieve acceptable AA results using ANOVA, indicating that genre is a strong independent variable that must be controlled for.

The literature outlined above all consider AA. There are two important works that investigate AV in specific, but not (Classical) Arabic AV, and both rely on negative data in some manner. [9] examines AV as a one-class classification problem, and use 'unmasking technique' to identify the rate at which accuracy of the classifier degrades in cross-validation between an unknown document and known documents of a given author. The authors propose that, compared to texts written by different authors, long texts written by the same author differ in only a small number of shared features. Eliminating such features makes it harder to differentiate between texts written by the same author, and lower accuracy of the classifier. This algorithm relies on negative data as part of the learning process (to compare the rate of decline of accuracy). [6] relies on a similarity metric inspired by [4]. It conducts a series of experiments using nine feature categories to determine the similarity between a given document and a corpus of documents of a given author, based on a Manhattan Distance measure. Their algorithm predicts verified attribution to a given author if the similarity value exceeds a certain threshold value θ, which is defined as the value where false negatives and false positives in the training set are equal. While this work does not rely on negative data for training, it still needs

[1] Unless the short texts (200 - 800 tokens in this experiment) are the complete work, using 'rare words' as a criterion may be problematic, as the structure of the text often determines where certain keywords appear in a given work.

negative training data to determine θ.

In this paper, I evaluate the possibility of dispensing with negative evidence altogether. Specifically, I show that θ in the experiment in [6] can be determined dynamically, entirely in terms of true positives, and lead to higher accuracy AV.

3. Corpus

In this section, I describe the content of the training and testing corpora, then I move to describing the formatting and preprocessing of the corpora. To mirror a typical AV task in medieval Arabic studies, a large number of complete works written by the same author are collected. In this case, 19 works attributed to Al-Ghazali are used for training. For testing positive results, the same 19 works are used via leave-one-out method. For testing negative results, a total of 12 works is used: nine classical works of authors belonging to the same time period, and from the same genres as Al-Ghazali; one text that has been proven to be falsely attributed to Al Ghazali using traditional methods (as in [13]); and two texts written in the twentieth Century (one fiction and one non-fiction). Table 1 shows the breakdown of the corpus used. The variation in the size and topic of the individual works is typical of the kind and size of data and problems faced by researchers in medieval Arabic studies, history, and literature. Classical texts were downloaded from two repositories widely used by scholars in Classical Arabic Studies: Shamela (www.shamela.ws) and Arabic Wikisource. The two modern Arabic texts were downloaded from book-cloud.com , which states that they observe relevant copyright laws. Selection of the texts for the corpus is primarily governed by practicality issues; Documents that are digitally available in text format, rather than PDF of images of manuscripts, are used.

3.1. Corpus Pre-processing, Formatting, and Feature Extraction

For pre-processing, short vowel diacritics, punctuation marks and numerals were removed. White spaces were normalized (e.g. line and paragraph breaks) to single spaces. Tokens are defined in this experiment as Arabic character strings separated by white spaces. Stemming is performed to the tokenized corpus using ISRI ([3] and [12]). Inspired by [6], the corpus is built into a number of problems, each problem P consists of a question document D, and a set of known documents S. In evaluating the algorithm, P is constructed such that S is the entire body of works of Al-Ghazaly, unless $D \in S$, then S = S - D.

4. Verification Method

In this section, I describe the training, testing, and evaluation methods. First, Tokenization is performed using NLTK regular expression tokenizer. Relative (normalized) frequency R for each token T is then calculated for each document D:

$$R = \frac{n_T}{n_D} \tag{1}$$

4.1. Training Procedure

Calculating document similarity:. Input to the training procedure is a set of documents with the same author. The output of the training procedure is a set of similarity values $S = S_1, S_2, S_3, ...S_n$, where $0 < S_n < 1$ represents the similarity of a training document D and the rest of the training corpus A.

Similarity is calculated, following [6], using the Manhattan Distance function between a questionable document X and a corpus of known documents Y:

$$dist(X, Y) = \Sigma_{j=1}^{n} |x_j - y_j| \tag{2}$$

Table 1. Corpus used

Corpus	Work	Size (1000 tokens)
Al-Gazaly	faḍā'iH al-bāṭiniyya	47
	aṣanāf al Maghrurīn	63.4
	mizān al-'amal	32.7
	al-tibr al-masbūk fī naṣīḥat al-mulūk	31.2
	Bidāyat al-hidāya	14.3
	Tahāfut al-falāsifa	49
	al-Wasīṭ fī al-madhab	400.7
	jawāhir al-Qur'ān	30.3
	iḥyā' 'ulūm al-dīn	831
	al-mustaṣfā min 'ilm al-uṣūl	181.7
	ma'ārij al-Quds fī madārij ma'rifat al-nafs	39.2
	al-Manḳūl min ta'līqāt al-uṣūl	53.4
	miškāt al-anwār	10.3
	miḥak al-naṭr fī al-mantiq	26.5
	mi'yār al-'ilm fī fann al-manṭiq	48.6
	qawā'id al-'aqā'id	18.7
	al-munqidh min al ḍalāl	11
	al-maqṣad al-'asnā fī šarḥ ma'ānī asmā' Allāh al-ḥusnā	34.2
	al-iqtiṣād fī al-i'tiqād	43.4
Others		
Falsely attributed to Ghazali	sirr il'ālimīn	22.3
ḳaṭīb al-baġdādi	šaraf aṣḥāb al-ḥadīth	23.1
	iqtiḍā' al-'ilm wa-al-'amal	13.3
ibn ḥazm al-andalusī	risalat al-radd 'ala al-kindi al-failasūf	10.1
	kitab al-imama wa al-mufadala (al-fiṣal fī al-milal wa-al-ahwā' wa-al-niḥal)	34
Ibn sīnā	al-Qānūn fī al-ṭibb	103.3
	kitāb al-siyāsa	46.1
ibn Rushd	Kitāb Faṣl al-maqāl wa-taqrīr mā bayna al-šarī'a wa-al-ḥikma min al-ittiṣāl	7.3
	Bidāyat al-mugtahid wa-nihāyat al-muqtaṣid	19.8
al-Qurtubi	al-I'lam bima fi din al-nasar ā min al-fasad wa-l-awham	4.8
Modern Texts		
Said Salem	Kaf Maryam (novel)	33.2
Afaf Abdel Moaty	Al mar'a wa al-sulta fi misr	43.4

Where x_j and y_j are the values for feature j in X and Y, in our case it is the relative (normalized) frequency of tokens. The next step is converting the resulting distance into a similarity score:

$$Sim(X, Y) = \frac{1}{1 + dist(X, Y)} \tag{3}$$

In essence, it possible to use other similarity measures (such as cosine similarity), as well as other possible ways of converting distance scores to similarity scores. I opt for using Manhattan Distance to minimize differences between this experiment and that used for the baseline. Any difference in accuracy is, then, due to the way a similarity threshold is determined, rather than other factors (as explained in the next subsection).

Determining similarity threshold:. The goal of this step is to determine what similarity value is a threshold θ that constitutes a cutoff point for accepting a suspect document as being written by the same author. Unlike approaches that consider false positives in calculating θ (c.f. section 2), I rely completely on the training data. To demonstrate the proposed method, I compare it to the method used to calculate the evaluation baseline [6] in 4.2. The baseline method uses Equal Error Rate (EER) to calculate θ. Under that approach, θ is identified as the similarity value where the rate

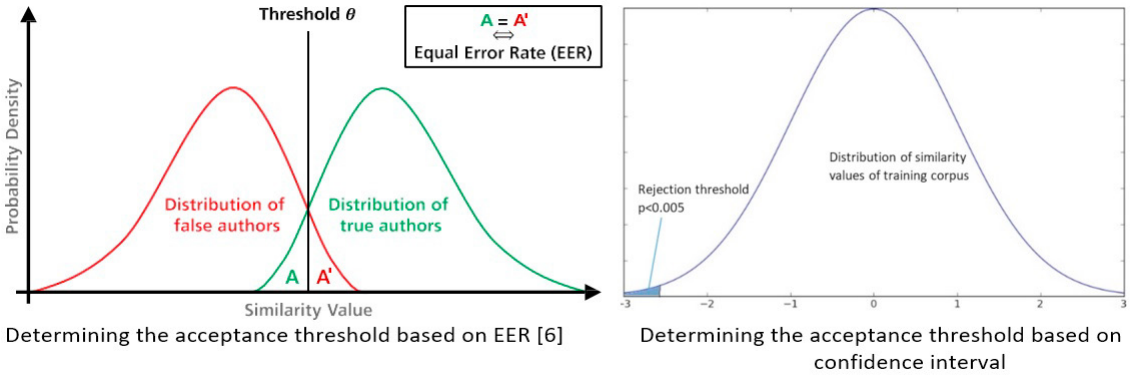

Fig. 1. Determining Theta.

of false positives (fraction of negative training data erroneously identified as genuine) is equal to the rate of false negatives (fraction of positive training problems erroneously identified as false attribution). The proposed method assumes that the similarity set S returned from the training data has normal distribution of similarity values,and requires no negative training data. Assuming normal distribution is reasonable, given the large set of feature values [2]. And if normal distribution is assumed, other standard parameters than EER can prove more efficient. For the proposed method, θ is defined as the similarity value of the lower bound of a confidence interval for the training set at $p < 0.005$. A candidate document D is accepted if its similarity value is higher than θ.

Testing:. classification procedure is performed to each problem as constructed in the previous section based on the training procedure. The output of the testing procedure is a list of answers (1, 0) regarding the attribution of question documents. This list is used for evaluation.

4.2. Evaluation baseline:

To evaluate the results returned by the experiment, I use as a baseline the accuracies reported by [6] on the best performance of the same feature category. Similar to the proposed algorithm, their approach measures similarity in terms of Manhattan Distance. Unlike the proposed algorithm, their approach uses negative training data to define θ based on EER. The authors report results on a number of languages other than Arabic (Table 2). I use the median accuracy reported for these languages as the first baseline.

Table 2. Baselines

Language	Accuracy
Dutch	62.12
English	65.1
Greek	57.37
Spanish	72.55
German	68.7
Median	**65.1**
Arabic EER	66.67

For Arabic-specific EER-based baseline, the positive training data is used together with 12 randomly selected negative evidence documents from the corpus to calculate accuracy for Arabic using EER-based method. Replicating

[2] In [6], the authors imply normal distribution of false and true authors (c.f. figure 1), but distribution is irrelevant in their determination of θ.

the experiment of [6] to the corpus at hand shows an accuracy of 66.67, which is used as the second baseline. This value is above the average accuracy reported for other languages, only lower than Spanish and German.

4.3. Experiment and Results

The goal of this experiment is to determine the extent to which a dynamically assigned similarity threshold can be used on positive evidence for AV. To achieve this, the proposed verification method is applied to each of the documents described in the Corpus section. To evaluate true positives, evaluation was implemented using leave-one-out method. The experiment was repeated 14 times with n = 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 15, 20, 25, 30. Accuracy of the outcome is calculated as the number of correctly classified problems divided by the total number of problems for each n. Table 3 shows the resulting accuracies across different values for n. As table 3 shows, the classifier is most accurate when considering similarity at the most common 3 - 9% tokens, with accuracy of 70.97%.

Table 3. Results

n%	**Accuracy (%)**
1	67.74
2	67.74
3	**70.97**
4	**70.97**
5	**70.97**
6	**70.97**
7	**70.97**
8	**70.97**
9	**70.97**
10	67.74
15	64.52
20	64.52
25	64.52
30	64.52

5. Evaluation and Discussion

As can be seen in Table 3 and figure 2, the proposed algorithm outperforms EER-based baselines in Arabic, at least when applied to a single class feature (best performance of n% most frequent tokens). It also performs better than most individual languages, with the exception of Spanish.

This exploratory experiment aims to be the basis for further investigation in the context of Classical Arabic authorship. As such, the construction of the corpus and the feature category used are different from previous literature used for the baseline or others dealing with Arabic data. For example, [6]'s work is based on much shorter documents from a variety of genres, and they are all of consistent size. It is often the case that researchers in Arabic studies have access to datasets like the corpus used here: very large documents of considerably variant document size. In the training set, the smallest document is 10,300 tokens, and the largest is 400,700+ tokens, with similar variation in the testing set. they also use feature categories that may not be accessible to researchers in Classical Arabic digital humanities, such as punctuation n-grams. Punctuation marks are often added and changed by researchers to the original manuscripts.

6. Conclusion and Future Work

I have presented a simple and effective method for AV that dispenses with the need for negative evidence. The method was tested using a single feature, the normalized token frequency of n% most frequent tokens, on a corpus of 31 documents to verify the authorship of Classical Arabic texts in related genres, written at the same time

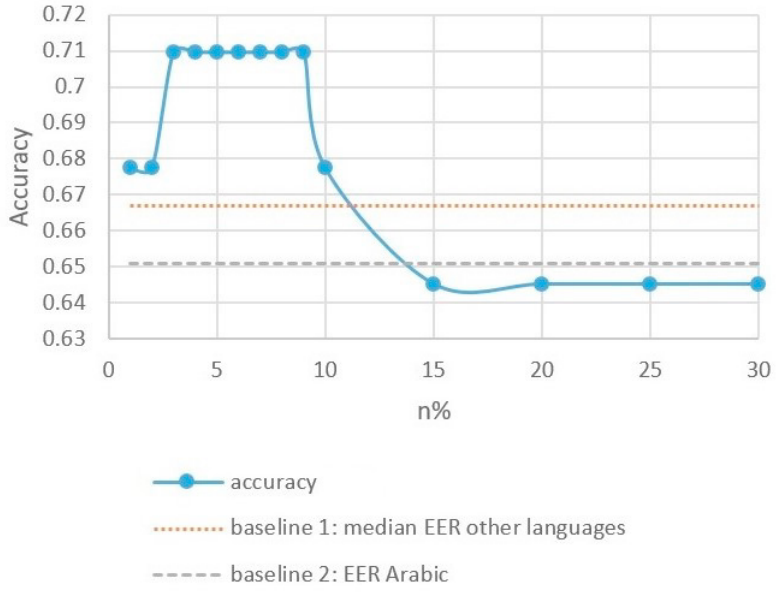

Fig. 2. Algorithm Performance.

period. The proposed method performs well compared to an EER-based method tested on the same corpus and reported in other languages. This method efficiently differentiates an AV task from an AA task, dispensing with potential inaccuracies related to the (un)availability of negative evidence. This paper focuses on testing the potential of the classification method. Future research investigates effective contexts where this method can be optimized.

Document size is a relevant variable. While in principle it is possible to use entire texts in the setting described in this experiment, it may be possible to increase computational efficiency by using only a portion of a 400,000-token book to reach a reliable authorship decision. Chunking texts in AV is relevant in two ways. First, chunking texts may affect how valuable certain features are in an AV task. Certain lexical items are more likely to appear at different parts of a document, which can affect the applicability of token frequency. Indeed, token frequency across chunks may still be valuable if the 'unit' of argument structure in Classical Arabic is at a chapter level or shorter, rather than on a whole-volume level. This is an empirical question. Second, AV across portions of a single text is also relevant when only a part of a given work is of questionable attribution, a common question in Arabic Studies. Another related factor is corpus size. Although in modern day corpus studies, this experiment uses a small corpus, it is representative of the kind of AA problem faced in Arabic Studies. Other AA problems in other fields (e.g. investigating web forums) have even smaller corpora. It has been noted by an anonymous reviewer that this method boils down to estimating the parameters of the normal distribution in figure 1. Indeed, the only parameter not determined based on the training data is the p value, which is arbitrarily set small enough to be acceptable for a large dataset. In essence, even p value can also be determined based on training data, albeit at a computational cost. Future research on smaller document size can offer valuable insights on that matter.

Future work also focuses on different feature types, lexical and syntactic, as well as examine the efficacy of the method in dealing with modern texts, and cross-linguistically. Initial investigation indicates that combining the proposed method with the least common, rather than most common, tokens and stems leads to improved performance. Due to the root-and-pattern nature of Arabic morphology, feature types typically used in European languages such as prefix and suffix n-grams may not be as effective when applied to Arabic corpora. In effect, suffixes and prefixes provide information about derivation and inflection in those languages. Arabic AV that investigates the role of morphological features in AV in Arabic needs to use alternative feature categories to the same effect.

References

[1] Abbasi, A., and Chen, H. (2005) "Applying authorship analysis to extremist-group Web forum messages." *IEEE Intelligent Systems*, **20**(5), 67–75. https://doi.org/10.1109/MIS.2005.81

[2] Altheneyan, A. S., and 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. https://doi.org/10.1016/j.jksuci.2014.06.006

[3] Bird, S., Klein, E., and Loper, E. (2009). "Natural language processing with Python: analyzing text with the natural language toolkit." *O'Reilly Media, Inc.*

[4] Burrows, J. (2002). " "Delta": a measure of stylistic difference and a guide to likely authorship." *Literary and Linguistic Computing*, **17**(3), 267–287. https://doi.org/10.1093/llc/17.3.267

[5] García-Barrero, D., Feria, M., and Turell, M. T. (2013). "Using function words and punctuation marks in Arabic forensic authorship attribution." In R. Sousa-Silva, R. Faria, N. Gavaldà, & B. Maia (Eds.), Proceedings of the 3rd European Conference of the International Association of Forensic Linguists (pp. 42–56). Porto, Portugal: Faculdade de Letras da Universidade do Porto.

[6] Halvani, O., Winter, C., and Pflug, A. (2016). "Authorship verification for different languages, genres and topics." *Digital Investigation*, **16**: S33–S43. https://doi.org/10.1016/j.diin.2016.01.006

[7] Howedi, F., and 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. Retrieved from http://iiste.org/Journals/index.php/CEIS/article/view/12132

[8] Kaddouri, S. (2000). "The attribution of al-i'lam bima fi din al-nasarā min al-fasad wa-l-awham to al-Qurtubi." *Al-Qantara*, **XXI**(1): 215–220.

[9] Koppel, M., & Schler, J. (2004). "Authorship verification as a one-class classification problem." *Twenty-first international conference on Machine learning - ICML '04*. https://doi.org/10.1145/1015330.1015448

[10] Ouamour, S., & Sayoud, H. (2013). "Authorship attribution of short historical Arabic texts based on lexical features." In *Proceedings - 2013 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, CyberC 2013* : 144–147. https://doi.org/10.1109/CyberC.2013.31

[11] Shaker, K. (2012). *Investigating features and techniques for Arabic authorship attribution.* Heriot-Watt University.

[12] Taghva, K., Elkhoury, R., and Coombs, J. (2005). "Arabic stemming without a root dictionary." In *International Conference on Information Technology: Coding and Computing, 2005. ITCC 2005*. (Vol. 1: 152-157). IEEE.

[13] Watt, W. M. (1952). "The Authenticity of the Works Attributed to al-Ghazālī." Journal of the Royal Asiatic Society of Great Britain and Ireland, **2**(1): 2445.