Tensor Space Models for Authorship Identification

Spyridon Plakias and Efstathios Stamatatos

Dept. of Information and Communication Systems Eng.
University of the Aegean
83200 – Karlovassi, Greece
stamatatos@aegean.gr

Abstract. Authorship identification can be viewed as a text categorization task. However, in this task the most frequent features appear to be the most important discriminators, there is usually a shortage of training texts, and the training texts are rarely evenly distributed over the authors. To cope with these problems, we propose tensors of second order for representing the stylistic properties of texts. Our approach requires the calculation of much fewer parameters in comparison to the traditional vector space representation. We examine various methods for building appropriate tensors taking into account that similar features should be placed in the same neighborhood. Based on an existing generalization of SVM able to handle tensors we perform experiments on corpora controlled for genre and topic and show that the proposed approach can effectively handle cases where only limited training texts are available.

Keywords: Authorship identification, Tensor space representation, Text categorization.

1 Introduction

Authorship identification is the task of assigning a text to an author, given a set of candidate authors for whom texts of undisputed authorship are available. Beyond the traditional approach based on human experts, this procedure can be automated by computational tools able to capture and match the stylistic properties of texts and authors [26, 32, 2]. The main idea is that by measuring some textual features we can distinguish between texts written by different authors. Nowadays, such tools are of increasing importance since there are plenty of texts in electronic form in Internet media (e.g., blogs, online forum messages, emails, etc.) indicating many applications of this technology, including forensics (identifying the authors of harassing email messages), intelligence (attributing messages to known terrorists), etc. [1, 23, 8, 33, 35]

One main issue in the research on authorship identification is the definition of appropriate textual features to quantify the stylistic properties of texts [13]. Many different measures have been proposed including simple ones such as word frequencies or character *n*-gram frequencies and more complex ones requiring some sort of syntactic or semantic analysis [35]. The other main issue is the development of attribution methodologies to assign texts to one candidate author. So far, the proposed

attribution models comprise standard discriminative algorithms (e.g., support vector machines) [9] and generative models (e.g., Bayesian methods) [27] as well as models specifically designed for authorship identification tasks [21, 29].

From a machine learning point-of-view, author identification can be viewed as a multi-class single-label text categorization (TC) task [28]. Actually, several studies on TC use this problem as just another testing ground together with other tasks, such as topic identification, language identification, genre detection, etc. [4, 25, 34] However, there are some points that make author identification a special TC task that should be handled carefully, namely:

Feature selection: Author identification is a style-based TC task. In such tasks, usually the most important features are the most frequent ones. On the contrary, in topic-based TC, the most frequent features (e.g., words) are usually excluded since they have little discriminatory power. Note that in case of word-features, the most frequent words carry no semantic information. In style-based tasks, such meaningless (or function) words are used unconsciously by the author, so they offer a way to measure their stylistic properties.

Shortage of training texts: In a typical author identification application only a few (possibly short) texts of undisputed authorship are available for the candidate authors. So, it is crucial for the attribution methodology to be able to effectively handle cases with low amount of training texts.

Class imbalance: It is not unusual to have an unequal distribution of training texts over the candidate authors. Note that beyond the amount of training texts per author, the length of training texts can also produce class imbalance conditions (in case we have short texts for some authors and long texts for other authors). In such cases, the evaluation procedure of an author identification approach should be carefully designed, especially in forensic applications. That is, the fact that there is shortage of training texts for one candidate author in comparison to the other authors does not mean that this person is less likely to be the true author of the text in question.

In this paper, we present an approach that attempts to take into account the aforementioned characteristics of the problem. In particular, we propose second order tensor space models for text representation in contrast to the traditional vector space models. The tensor models are able to handle the same amount of textual features with the vector models but require much fewer parameters to be learnt. Therefore, they are suitable for cases with limited training data. On the other hand, in contrast to vector models, the positioning of each feature into the tensor model plays a crucial role since relevant features should be placed in the same neighborhood. To solve this problem, we propose a frequency-based metric to define the relevance between features and explore several methods for filling the feature matrix. Using an existing generalization of the SVM algorithm able to handle tensors of second order [6] we perform experiments on text corpora controlled for genre and topic under balanced and imbalanced conditions. In the latter case, we pay special attention to the

evaluation methodology so that the test corpus distribution over the authors is uncorrelated with the corresponding distribution of the training corpus.

The rest of this paper is organized as follows. Section 2 includes previous work in author identification while Section 3 describes in detail the proposed tensor space models. Section 4 describes the conducted experiments and, finally, Section 5 summarizes the conclusions drawn by this study and proposes future work directions.

2 Previous Work

The majority of the work in authorship identification (or authorship attribution) deals with defining appropriate measures for quantifying the writing style. This line of research is known as *stylometry* [13]. Several hundreds of stylometric features have been proposed attempting to find measures that are reliable and accurate under varying text-length, text types, and availability of text processing tools. The most traditional features are word-based in accordance to work in topic-based text categorization. However, in author identification, the most frequent words have been proved to be the most important ones [5]. Such words, including articles, prepositions, conjunctions, ('stop words' in information retrieval terminology) are usually excluded from topic-based classification since they carry no semantic information. On the other hand, they are closely related to certain syntactic structures. That is why they are also called 'function' words. So, their use indicates the use of certain syntactic structures by the author. Several sets of function words have been defined for English [2, 3, 20]. An alternative way to automatically define the function word set is to extract the most frequent words in a corpus [24, 29]. There are also attempts to use word *n*-grams to exploit contextual information [27, 7]. However, this process considerably increases the dimensionality of the problem and has not produced encouraging results so far.

Another way to represent text is by using character *n*-gram frequencies [17, 30]. Again the most frequent character *n*-grams (*n* contiguous characters) include the most important information. Although the dimensionality of the problem is increased in comparison to a function word approach, it is much smaller in comparison to a word *n*-gram approach. Methods based on such features have produced very good results in several author identification experiments and texts in various languages [17, 16, 30, 11]. However, there is still no consensus about the definition of an appropriate *n* value (the length of character *n*-grams) for certain natural languages and text types. Another character-based approach makes use of existing text compression tools to estimate the similarity of texts [4, 25].

A more elaborate type of features involves the use of natural language processing tools to extract syntactic [32, 10, 12] or semantic information [3, 10]. In theory, such features should better quantify the stylistic choices of the authors since they are used unconsciously. However, the measurement of such features in raw text is still a difficult procedure and the extracted measures are noisy. As a result, the quantification of writing style is not accurate enough.

Beyond the definition of stylometric features, the research in author identification is dominated by the development of effective attribution methodologies. A significant part of the studies is based on discriminative models utilizing machine learning

techniques like SVM [9, 21, 23, 33], neural networks [35], ensemble methods [29] etc. Such powerful machine learning algorithms can effectively cope with high dimensional and sparse data. Another approach is to apply a generative model, like a naïve Bayes model [27]. Yet another approach is to estimate the similarity between two texts [4, 17].

Recently, a number of studies take into account factors such as training set size, imbalanced training data, and the amount of candidate authors in order to build more reliable author identification methods. Marton. et al. [25] and Hirst & Feiguina [12] examine the effectiveness of author identification methods under limited training text conditions. Stamatatos [30] proposes a model for handling limited and imbalanced training texts. In another study, Stamatatos [31] proposes text sampling methods for re-balancing an imbalanced training corpus to improve author identification performance. Finally, Madigan, et al. [24] test various author identification methods in cases with high number of candidate authors.

3 Tensor Space Representation

In a vector space model, a text is considered as a vector in R^n, where n is the number of features. A second order tensor model considers a text as a matrix in $R^x \otimes R^y$, where x and y are the dimensions of the matrix. A vector $\mathbf{x} \in R^n$ can be transformed to a second order vector $\mathbf{X} \in R^x \otimes R^y$ provided $n \approx x*y$. Notice that the same features are used in both the vector and the tensor model.

A linear classifier in R^n (e.g., SVM) can be represented as $\mathbf{a}^T\mathbf{x}+b$, that is, there are $n+1$ parameters to be learnt (b, a_i, $i=1,\ldots,n$). Similarly, a linear classifier in $R^x \otimes R^y$ can be represented as $\mathbf{u}^T\mathbf{X}\mathbf{v}+b$, that is, there are $x+y+1$ parameters to be learnt (b, u_i, $i=1,\ldots y$, v_j, $j=1,\ldots x$). Consequently, the number of parameters is minimized when $x=y$ (i.e., square matrix) and this is much lower than n. Therefore, the vector space representation is more suitable in cases with limited training sets since much fewer parameters have to be learnt. Note that in both cases the amount of textual feature is the same (n). However, in the tensor model the position of each feature within the matrix plays a crucial role for the performance of the model since each feature is strongly associated with the features of the same row or column. On the other hand, the position of a feature in the vector model does not affect the performance of the model.

3.1 Support Tensor Machines

To be able to handle tensors instead of vectors, we use a generalization of SVM, called *support tensor machines* (STM) [6]. Initially, this algorithm sets $\mathbf{u}=(1,\ldots,1)^T$ and uses it to compute the initial $\mathbf{v}$. Then, it works iteratively by computing in each step a new $\mathbf{u}$ and $\mathbf{v}$ as follows (given a set of training examples $\{\mathbf{X}_i, y_i\}$, $i=1,\ldots,m$):

Computation of $\mathbf{v}$ (provided $\mathbf{u}$) solving the following optimization problem:

$$\min_{\mathbf{v},b,\xi} \frac{1}{2}\|\mathbf{u}\|^2 \mathbf{v}^T\mathbf{v} + C\sum_{i=1}^{m}\xi_i$$
$$\text{subject to } y_i(\mathbf{v}^{\mathrm{T}}\mathbf{X}_i^T\mathbf{u} + b) \geq 1-\xi_i, \xi_i \geq 0, i = 1,...,m \quad (1)$$

Note that this optimization problem is the same as the standard SVM algorithm. So, any computation method used in SVM can also be used here.

Computation of **u** (provided **v**) solving the following optimization problem:

$$\min_{\mathbf{v},b,\xi} \frac{1}{2}\|\mathbf{v}\|^2 \mathbf{u}^T\mathbf{u} + C\sum_{i=1}^{m}\xi_i$$
$$\text{subject to } y_i(\mathbf{u}^{\mathrm{T}}\mathbf{X}_i^T\mathbf{v} + b) \geq 1-\xi_i, \xi_i \geq 0, i = 1,...,m \quad (2)$$

Again, this optimization problem is the same as the standard SVM algorithm and any computation method used in SVM can also be used here. The procedure of calculating new values for **u** and **v** is repeated until they tend to converge.

3.2 Feature Relevance

Since the tensor-based model takes into account associations between features (each feature is strongly associated with features in the same row and column) it is crucial to place relevant features in the same neighbourhood. To suitably transform a vector representation to a second order tensor representation, one has to define what features are considered relevant and how relevant features are placed in the same neighbourhood.

As it has been demonstrated by several authorship identification studies, the frequency of features is a crucial factor for their significance [14, 19]. Actually, the frequency information is more important than the discriminatory power of the features when examined individually. Following this evidence, in this paper, we use the frequency of occurrence as the factor that determines relevance among features. Particularly, in a binary classification case, where we want to discriminate author A from author B, the relevance $r(x_i)$ of a feature x_i is:

$$r(x_i) = \frac{f_A(x_i) - f_B(x_i)}{f_A(x_i) + f_B(x_i) + b} \quad (3)$$

where $f_A(x_i)$ and $f_B(x_i)$ are the relative frequencies of occurrence of feature x_i in the texts of author A and B, respectively, and b a smoothing factor. The higher the $r(x_i)$, the more important the feature x_i for author A. Similarly, the lower the $r(x_i)$, the more important the feature x_i for author B. Note, that the relevance metric is not necessarily associated with the discriminatory power of each feature. However, a feature with high (low) relevance value is likely to be a good discriminator for author A (B) since it is found more times in their texts in comparison to author B (A).

$$\begin{bmatrix} P1 & P2 \\ P3 & P4 \end{bmatrix}$$

P1 comprises features strongly associated with author A
P4 comprises features strongly associated with author B
P2 and P3 comprise neutral features

Fig. 1. A second order tensor divided into four parts according to the feature relevance.

3.3 Matrix Filling

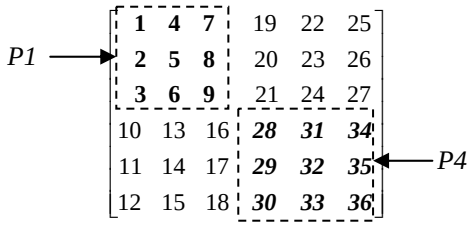
Given a ranking of features according to the relevance metric, we need a strategy to fill the matrix of features having in mind that relevant features should be placed in the same neighbourhood. To this end, we consider that each matrix is segmented into four parts of equal size, as it is depicted in figure 1. The upper left part (*P1*) is filled with features strongly associated with author A, the lower right part (*P4*) is filled with features strongly associated with author B, while the two remaining parts (*P2* and *P3*) are filled with relatively neutral features. So, we attempt to create a distance between the features strongly associated with one of the authors in both rows and columns of the feature matrix. Moreover, each row or column of the matrix is strongly associated with one of the authors since it contains some very relevant features for that author and a number of neutral features. That is, the rows and columns of the matrix are composed of a combination of features from *P1* and *P2* or *P3* as well as a combination of features from *P4* and *P2* or *P3*. On the other hand, there are no rows or columns of the matrix that contain features strongly associated with both authors (that is, a combination of features from *P1* and *P4* is not allowed).

To place each feature within each part of the matrix we fill the columns of that part of the matrix from left to right with decreasing relevance values. As a result, the columns of the matrix are filled with features of similar relevance values, while the rows are filled with features of mixed relevance values. This is depicted in figure 2a. We call this matrix filling approach *symmetric cross*.

A variation of this technique is to fill each part of the matrix diagonally. In more detail, we start from the upper left corner of each part of the matrix and fill the diagonals with decreasing relevance values, as it is shown in figure 2b. This method distributes the most significant features in a fairer way across the rows and columns of the matrix. We call this matrix filling method *cross-diagonal*.

Yet another variation of the symmetric cross method is to segment the feature matrix into four parts of different size. That is, the upper left part may be smaller than the lower right part of the matrix (see figure 2c). This may correspond to the cases where only a few features are strongly associated with author A while most of the features are associated with author B. The boundaries of the parts of the matrix can be found by examining the relevance values given that positive relevance values indicate features important for author A and negative relevance values indicate features important for author B. We call this matrix filling method *asymmetric cross*.

(a)



1	**4**	**7**	19	22	25
2	**5**	**8**	20	23	26
3	**6**	**9**	21	24	27
10	13	16	***28***	***31***	***34***
11	14	17	***29***	***32***	***35***
12	15	18	***30***	***33***	***36***

(b)

1	**3**	**6**	19	21	24
2	**5**	**8**	20	23	26
4	**7**	**9**	22	25	27
10	12	15	***28***	***30***	***33***
11	14	17	***29***	***32***	***35***
13	16	18	***31***	***34***	***36***

(c)

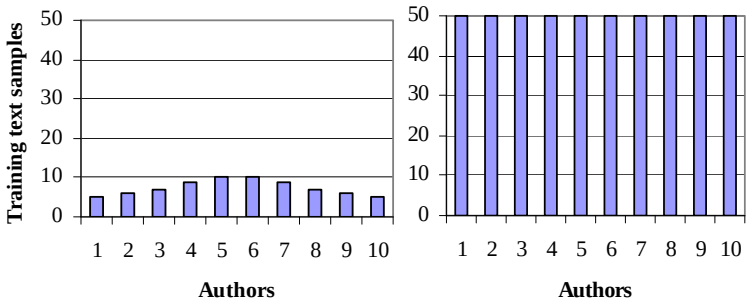
1	**3**	13	15	17	19
2	**4**	14	16	18	20
5	9	***21***	***25***	***29***	***33***
6	10	***22***	***26***	***30***	***34***
7	11	***23***	***27***	***31***	***35***
8	12	***24***	***28***	***32***	***36***

Fig. 2. Examples of matrix filling with the proposed techniques. The feature numbers correspond to the ranking of 36 features according to their relevance (1 correspond to the feature with higher relevance value). Features in boldface (*P1*) are strongly associated with author A, features in boldface italics are strongly associated with author B (*P4*). (a) Symmetric cross: the four parts of the matrix are of equal size and the columns of each part are filled with features of decreasing relevance from the left to the right. (b) Cross-diagonal: the four parts of the matrix are of equal size and each part is filled with features of decreasing relevance from upper left corner to the lower right corner. (c) Asymmetric cross: the four parts of the matrix are of different size and the columns of each part are filled with features of decreasing relevance from the left to the right.

4 Experiments

4.1 Corpus and Settings

The corpora used for evaluation in this study consist of newswire stories in English taken from the publicly available Reuters Corpus Volume 1 (RCV1) [22]. Although this corpus was not particularly designed for evaluating author identification methods,



(a) (b)

Fig. 3. (a) A 5:10 imbalanced corpus of 10 authors comprising at least 5 texts per author and a maximum of 10 texts for some authors. (b) A balanced corpus of 10 authors comprising 50 texts per author.

it offers a large pool of texts of unquestioned authorship that cover a variety of topics and it has already been used by previous studies [24, 18, 31]. We selected the top 10 authors with respect to the amount of texts belonging to the topic class CCAT (about corporate and industrial news) to minimize the topic factor for distinguishing between texts. Given that this corpus is already controlled for genre, we expect the authorship factor to be the most important discriminating factor.

We produced several versions of this corpus by varying the amount of training texts per author. To produce three balanced training corpora, we used 50, 10 or 5 training texts per author, respectively. In all cases, the test corpus comprises 50 texts per author not overlapping with the training texts (see figure 3b).

To produce imbalanced training corpora we applied a Gaussian distribution over the authors. In particular, we set the minimum and maximum amount of training texts per author and an imbalanced Gaussian distribution defines the amount of training texts per author as it is depicted in figure 3a. Three imbalanced training corpora were used, 10:20, 5:10, and 2:10 where the notation *a*:*b* means that at least *a* training texts are available for all the authors and *b* is the maximum amount of training texts per author. The test corpus comprises 50 texts per author not overlapping with the training texts and it is the same with the test corpus of the balanced training corpora. Note that, by using a balanced test corpus when we know that the training corpus is imbalanced, we attempt to simulate the general authorship identification case where the availability of many training texts for one author should not increase their probability to be the true author of the unknown texts.

4.2 Results

To represent the texts we used a character *n*-gram approach. Thus, the feature set consists of the 2,500 most frequent 3-grams of the training corpus. This means that a standard linear SVM model [15] can be built using vectors of 2,500 features. Moreover, a second order tensor model can be built based on a 50x50 matrix. Note

that since we deal with a multi-class author identification task, we followed a *one vs. one* approach, that is, for each pair of authors a STM model was built and the matrix filling technique was based on the feature relevance for that pair of authors. Based on preliminary experiments, we set the *C* parameter of linear SVM to 1, the corresponding parameter for STM models to 0.1 and the smoothing parameter *b* equal to 1.

We tested the STM models with the matrix filling techniques proposed in section 3.3 and compare it to the standard SVM model. The performance results for both the balanced and imbalanced training corpora can be seen in table 1. Recall, that in all the cases the test corpus is the same so these results are directly comparable. The method called STM-Simple is based on a very naïve methodology for filling the feature matrix: the features are ranked in decreasing frequency and, then, the rows of the matrix are filled from the top row to the last raw and from left to right. Therefore, the comparison of the techniques proposed in section 3.3 with this simple method reveals the significance of taking into account the relevance of the features when filling the matrix. As can be seen, the proposed techniques achieve clearly better performance results in comparison to this simple baseline method. So, it is crucial to place similar (relevant) features in the same neighbourhood when filling the feature matrix of an STM model.

Comparing SVM and STM models for the balanced cases reveals that the standard SVM model outperforms STM models when many training texts (50 per author) are available. On the other hand, when the balanced training corpus is limited (10 or 5 texts per author) all the examined STM models are better than SVM. This confirms our hypothesis that the STM can more effectively handle limited training data since much fewer parameters have to be learnt. The imbalanced cases produce more confused results. In two cases the SVM model is the clear winner while in the third case an STM model slightly outperforms the SVM model.

Table 1. Classification accuracy (%) of the SVM and the proposed STM models with various matrix filling techniques.

Model	Training texts per author					
	Balanced			Imbalanced		
	50	10	5	10:20	5:10	2:10
SVM	**80.8**	64.4	48.2	**64.2**	62.4	**51.0**
STM-Simple	70.4	54.4	44.2	58.2	49.2	39.0
STM-Symmetric cross	76.6	**67.8**	50.4	63.0	59.8	49.2
STM-Cross-diagonal	76.0	64.4	52.4	62.2	**62.6**	49.8
STM-Asymmetric cross	78.0	65.2	**53.4**	61.8	60.6	50.0

5 Conclusions

In this paper, we presented an approach to author identification that is based on a tensor space representation instead of the traditional vector space model. The proposed representation can be used in a classification scheme that requires much fewer parameters to be learnt and it is more suitable in cases where only limited

training data are available. Author identification is a representative example of this type of problems where usually extremely limited texts of known authorship are available, especially in forensic applications. A generalization of the SVM algorithm able to handle second order tensors was used. The conducted experiments have shown the effectiveness of the proposed model in cases of shortage of training texts in comparison to a standard SVM model.

A consequence of the second order tensor representation is that the position of each feature within the matrix plays a crucial role since features of the same row or column are strongly associated. To place relevant features in the same neighborhood of the matrix, we first defined a relevance metric that is based on frequency information and then examined various matrix filling techniques. The comparison of the proposed techniques to a baseline method revealed that the information we used is very important for achieving good results. On the other hand, it is not clear which of the proposed techniques is superior. Further experiments should be conducted towards this direction.

The performed experiments were based on both balanced and imbalanced training corpora. However, in all cases the test corpus was balanced to extract more reliable evaluation results since in a typical author identification scenario, the existence of many texts of undisputed authorship for one candidate author should not increase the likelihood of being the true author of the unknown text. Although the proposed models managed to increase the performance (in comparison to a standard vector space SVM model) when dealing with limited training texts, their performance in the imbalanced cases was not encouraging. More elaborated matrix filling techniques and possibly a different definition of feature relevance should be tested for effectively handling imbalanced training corpora in author identification tasks.

References

1. Abbasi, A., Chen, H.: Applying Authorship Analysis to Extremist-group Web Forum Messages. IEEE Intelligent Systems, 20(5), 67--75 (2005)
2. Argamon, S., Saric, M., Stein, S.: Style Mining of Electronic Messages for Multiple Authorship Discrimination: First Results. In: 9th ACM SIGKDD, pp. 475--480 (2003)
3. Argamon, S., Whitelaw, C., Chase, P., Hota, S.R., Garg, N., Levitan, S.: Stylistic Text Classification Using Functional Lexical Features. Journal of the American Society for Information Science and Technology, 58(6), 802--822 (2007)
4. Benedetto, D., Caglioti, E., Loreto, V.: Language Trees and Zipping. Physical Review Letters, 88(4), 048702 (2002)
5. Burrows, J.F.: Word Patterns and Story Shapes: The Statistical Analysis of Narrative Style. Literary and Linguistic Computing, 2, 61--70 (1987)
6. Cai, D., He, X., Wen, J.R., Han, J., Ma, W.Y.: Support Tensor Machines for Text Categorization. Technical report, UIUCDCS-R-2006-2714, University of Illinois at Urbana-Champaign (2006)
7. Coyotl-Morales, R.M., Villaseñor-Pineda, L., Montes-y-Gómez, M., Rosso, P.: Authorship Attribution Using Word Sequences. In: 11th Iberoamerican Congress on Pattern Recognition, pp. 844—853, Springer, (2006)d

8. Chaski, C.E.: Who's at the Keyboard? Authorship Attribution in Digital Evidence Investigations. International Journal of Digital Evidence, 4(1) (2005)
9. Diederich, J., Kindermann, J., Leopold, E., Paass, G.: Authorship Attribution with Support Vector Machines. Applied Intelligence, 19(1/2), 109--123 (2003)
10. Gamon, M.: Linguistic Correlates of Style: Authorship Classification with Deep Linguistic Analysis Features. In: 20th International Conference on Computational Linguistics, pp. 611--617 (2004)
11. Grieve, J.: Quantitative Authorship Attribution: An Evaluation of Techniques. Literary and Linguistic Computing, 22(3), 251--270 (2007)
12. Hirst, G. Feiguina, O.: Bigrams of Syntactic Labels for Authorship Discrimination of Short Texts. Literary and Linguistic Computing, 22, 405--417, Oxford University Press, (2007)
13. Holmes, D.I.: The Evolution of Stylometry in Humanities Scholarship. Literary and Linguistic Computing, 13(3), 111--117 (1998)
14. Houvardas, J., Stamatatos E.: N-gram Feature Selection for Authorship Identification. In: 12th International Conference on Artificial Intelligence: Methodology, Systems, Applications, pp. 77--86, Springer, (2006)
15. Joachims, T.: Text Categorization with Support Vector Machines: Learning with Many Relevant Features. In: 10th European Conference on Machine Learning, pp. 137--142 (1998)
16. Juola, P.: Authorship Attribution for Electronic Documents. In: M. Olivier and S. Shenoi (eds.) Advances in Digital Forensics II, pp. 119--130, Springer (2006)
17. Keselj, V., Peng, F., Cercone, N., Thomas, C.: N-gram-based Author Profiles for Authorship Attribution. In: Pacific Association for Computational Linguistics, pp. 255--264 (2003)
18. Khmelev, D.V., Teahan, W.J.: A Repetition based Measure for Verification of Text Collections and for Text Categorization. In: 26th ACM SIGIR, pp. 104--110 (2003)
19. Koppel, M., Akiva, N., Dagan, I.: Feature Instability as a Criterion for Selecting Potential Style Markers. Journal of the American Society for Information Science and Technology, 57(11), 1519--1525 (2006)
20. Koppel, M., Schler, J.: Exploiting Stylistic Idiosyncrasies for Authorship Attribution. In: IJCAI'03 Workshop on Computational Approaches to Style Analysis and Synthesis, pp. 69-72 (2003)
21. Koppel, M., Schler, J., Bonchek-Dokow, E.: Measuring Differentiability: Unmasking Pseudonymous Authors. Journal of Machine Learning Research, 8, 1261--1276 (2007)
22. Lewis, D., Y. Yang, T. Rose, F. Li.: RCV1: A New Benchmark Collection for Text Categorization Research. Journal of Machine Learning Research, 5, 361--397 (2004)
23. Li, J., Zheng, R., Chen, H.: From Fingerprint to Writeprint. Communications of the ACM, 49(4), 76--82 (2006)
24. Madigan, D., Genkin, A., Lewis, D., Argamon, S., Fradkin, D., Ye, L.: Author Identification on the Large Scale. In: CSNA-05 (2005)
25. Marton, Y., Wu, N., Hellerstein, L.: On Compression-based Text Classification. In: European Conference on Information Retrieval, pp. 300--314, Springer (2005)
26. Mosteller, F., Wallace, D.: Applied Bayesian and Classical Inference: The Case of the Federalist Papers. Addison-Wesley, Reading, MA (1964)
27. Peng, F., Shuurmans, D., Wang, S.: Augmenting Naive Bayes Classifiers with Statistical Language Models. Information Retrieval Journal, 7(1), 317--345 (2004)

28. Sebastiani, F.: Machine Learning in Automated Text Categorization. ACM Computing Surveys, 34(1) (2002)
29. Stamatatos, E.: Authorship Attribution Based on Feature Set Subspacing Ensembles. International Journal on Artificial Intelligence Tools, 15(5), 823--838 (2006)
30. Stamatatos, E.: Author Identification Using Imbalanced and Limited Training Texts. In: 4th International Workshop on Text-based Information Retrieval, pp. 237--241 (2007)
31. Stamatatos, E.: Author Identification: Using Text Sampling to Handle the Class Imbalance Problem. Information Processing and Management, 44(2), 790--799 (2008)
32. Stamatatos, E., Fakotakis, N., Kokkinakis, G.: Automatic Text Categorization in Terms of Genre and Author. Computational Linguistics, 26(4), 471--495 (2000)
33. de Vel, O., Anderson, A., Corney, M., Mohay, G.: Mining E-mail Content for Author Identification Forensics. SIGMOD Record, 30(4), 55--64 (2001)
34. Zhang, D., Lee, W.S.: Extracting Key-substring-group Features for Text Classification. In: 12th Annual SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 474--483 (2006)
35. Zheng, R., Li, J., Chen, H., Huang, Z.: A Framework for Authorship Identification of Online Messages: Writing Style Features and Classification Techniques. Journal of the American Society of Information Science and Technology, 57(3), 378--393 (2006)