Journal of Intelligent & Fuzzy Systems xx (20xx) x–xx
DOI:10.3233/JIFS-179899
IOS Press



# Authorship attribution of Spanish poems using n-grams and the web as corpus

Rafael Guzmán-Cabrera*
*Department of Electrical Engineering, Engineering Division, Irapuato-Salamanca Campus, University of Guanajuato, Mexico*



**Abstract**. In many areas of professional development, the categorization of textual objects is of critical importance. A prominent example is the attribution of authorship, where symbolic information is manipulated using natural language processing techniques. In this context, one of the main limitations is the necessity of a large number of pre-labeled instances for each author that is to be identified. This paper proposes a method based on the use of n-grams of characters and the use of the web to enrich the training sets. The proposed method considers the automatic extraction of the unlabeled examples from the Web and its iterative integration into the training data set. The evaluation of the proposed approach was done by using a corpus formed by poems corresponding to 5 contemporary Mexican poets. The results presented allow evaluating the impact of the incorporation of new information into the training set, as well as the role played by the selection of classification attributes using information gain.

Keywords: Authorship attribution, self-training, web corpora

## 1. Introduction

Classification refers to the task of assigning a set of objects to two or more predefined categories. In many areas of professional development, such as the attribution of authorship, the categorization of new objects is of critical importance. Unfortunately, in most cases this process is expensive and time-consuming. Thus, there is an avid interest in the development of new technologies and approaches which can achieve an automatic classification, especially for the case of textual objects [1].

A typical approach to build a text categorization system in general consists in manually assigning a set of documents to categorize. In this case the hierarchies or thematic areas are assigned by an expert. However, this process is usually very expensive, since there is a need for an expert for each area or application for which the classification is to be carried out; also, a change of area requires new experts to define the categories or documents that belong to each category as well as the rules that allow decisions on new documents to be classified [2]. Because of this, the most commonly used approach nowadays is to use information retrieval techniques and machine learning to produce a classification model [3–5]. Learning-based systems are also faster to build with respect to rules-based systems or language models.

Authorship attribution is the task of identifying the author of a given text. It can be considered as a classification problem, where a set of documents with known authorship are used for training, and the aim is to automatically determine the corresponding author of an anonymous text.

Applications of authorship attribution include plagiarism detection (i.e. college essays), deducing the writer of inappropriate communications that were

*Corresponding author. Rafael Guzmán Cabrera, Department of Electrical Engineering, Engineering Division, University of Guanajuato Mexico. E-mail: guzmanc@ugto.mx.

ISSN 1064-1246/20/$35.00 

sent anonymously or under a pseudonym (i.e. threatening or harassing e-mails), as well as resolving historical questions of unclear or disputed authorship. Specific examples are the Federalist papers [6] and the forensic analysis of the Unabomber manifesto [7].

In this paper presents the application of a web-based self-training method in a non-thematic classification task, namely, authorship attribution. In order to evaluate the performance of my approach in severe conditions, I focus the experiments on poem classification where documents are usually short and both their vocabulary and structure can differ significantly from predominant web language.

The rest of the paper is organized as follows: in section 2 a brief state of the art of attribution of authorship is presented. In section 3 the methodology implemented in this work is presented in detail. The result obtained are presented and, finally, the conclusions and future work are outlined.

## 2. Authorship attribution

The first programs for similarity and plagiarism detection were developed by Halstead [8] and McCabe [9]. Subsequently, the Wise program [10] was developed to compare line by line using Levenshtein's distance (chain similarity). Then, Gitchell and Tran [11] worked based on the analysis of the tree generated by a program. More recent works [12] also use latent semantic analysis (LSA). And, even more recently, some proposals have been implemented to detect similarity in computer programs, for example, in Karel language [13]. An example is the "Moss" program [14], which compares two programs according to the similarity of their "fingerprints."

The measure of cosine or its modification, the soft cosine [15], which allows taking into account the similarity between characteristics in a vector space model, is also often used to measure similarity. In the present work, on the other hand, I note that the use of n-grams (sequences of words or characters) as characteristics for classification is actually suitable in tasks related to machine learning for texts. There are other options such as syntactic n-grams [16], integrated syntactic graphs [17], tree editing distance [18], among others, which can be used in the vector space model [19].

In contrast to previous works, this paper does not propose another document representation for authorship attribution, instead it describes a new semi-supervised learning method that allows working with small training sets. As expected, my web-based self-training classification method may be applied along with all these kinds of features. However, given that my interest is to have a general approach for authorship attribution that allow analyzing documents of different sizes and domains, I have decided to mainly explore the use of word-based features, in particular, *n*-grams.

## 3. Methodology proposed

Given that there is not a standard data set for evaluating authorship attribution methods, I had to assemble my own corpus. I have a corpus formed by 353 poems of 5 contemporary Mexican poets namely Efraín Huerta, Jaime Sabines, Octavio Paz, Rosario Castellanos, and Rubén Bonifaz. Everyone with particularly unique writing style. Table 1 resumes some statistics about this corpus.

In general, the corpus comprises short poems with only 20 sentences per poem on average; the number of words per poem is 175, while the number of different words in each one is 57 on average; and, again, all of them correspond to Mexican contemporary poets. In particular, I was very careful on selecting modern writers in order to avoid the identification of authors by the use of anachronisms. Figure 1 shows the general scheme of my semi-supervised text classification method. It consists of two main processes. The first one deals with the corpora acquisition from the Web,

Table 1
Corpus statistics

| Poets | Number of documents | Word forms | Word tokens | Word forms (in Training Set) | Phrases by Document |
|---|---|---|---|---|---|
| Efraín Huerta | 48 | 3831 | 11352 | 2827 | 22.3 |
| Jaime Sabines | 80 | 3955 | 12464 | 2749 | 17.4 |
| Octavio Paz | 75 | 3335 | 12195 | 2431 | 27.2 |
| Rosario Castellanos | 80 | 4355 | 11944 | 3280 | 16.4 |
| Rubén Bonifaz | 70 | 4769 | 12481 | 3552 | 17.3 |

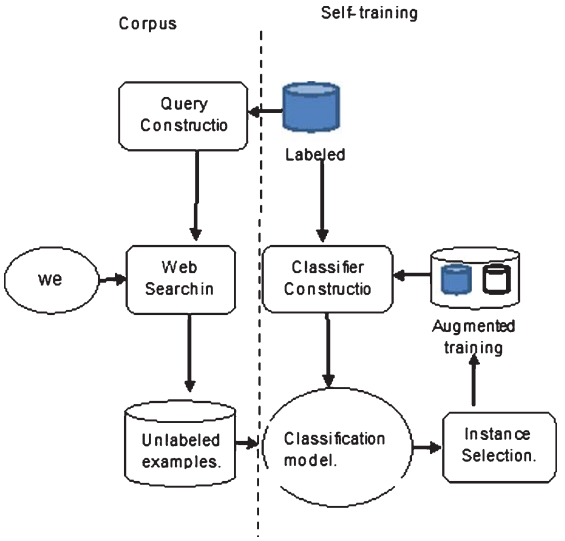


Fig. 1. General overview of the classification method.

whereas the second one focuses on the self-training learning approach [20].

The Corpora Acquisition process considers the automatic extraction of unlabeled examples from the Web. In order to do this, it first constructs a number of queries by combining the most significant words for each class; then, using these queries, it looks at the Web for some additional training examples related to the given classes. The Query at the web where constructed according to the method shown in [21].

For the development of this work, two classification methods were used: Bayes and Support vector machines (SVM). The Bayesian classifier is considered as part of the probabilistic classifiers, which are based on the assumption that interest amounts are governed by probability distributions, and that the optimal decision can be made by reasoning about those probabilities along with the observed data. The Naive Bayes algorithm uses the training set to estimate the parameters of a probability distribution that describes the training set. The category with the highest probability is the assigned category. On the other hand, in geometric terms, SVM can be seen as the attempt to find a surface ($\sigma_i$) that separates positive examples from negative ones by the widest possible margin. The search for $\sigma_i$ that meets the minimum distance between it and an example of training is maximum, is performed across all surfaces $\sigma_1, \sigma_2, \ldots$ in the A-dimensional space that separate the positive examples from the negatives in the training set (known as decision surface). The best decision surface is determined only by a small set of training examples, called **support vectors**. An important advantage of SVM is that they allow constructing non-linear classifiers, that is, the algorithm represents non-linear training data in a space of high dimensionality (called "characteristic space"), and builds the hyperplane that has the maximum margin. In addition, it is possible to calculate the hyperplane without explicitly representing the feature space.

## 4. Obtained results

An experiment was designed that allowed us to appreciate mainly two things. First, the impact that classification accuracy has on the incorporation of new unlabeled information, coming from the web, to the training set through an iterative process. Second, the performance of the classification systems when making the selection of characteristics that will be used as classification attributes (IG> 0).

For the evaluation of the experiment, two standard classification methods were used namely Bayes and SVM. Accuracy and recall were used as performance evaluation measures.

The corpus was divided in two data sets: training (with 80% of the labeled examples) and test (with 20% of the examples). The idea was to carry out the experiment in an almost-real situation, where it is not possible to know in advance all the vocabulary. This is a very important aspect to take into account in poem classification since poets tend to employ a very rich vocabulary.

In this work, I used *n*-grams as document features. I mainly performed two different experiments. In the first one I used bigrams as features, whereas in the second one I used trigrams. In each case, experiments were performed calculating the information gain with the purpose of having a comparison of the impact of the information gain on the accuracy of classification.

Table 2 shows the results corresponding to the first five iterations of the method. As can be observed, the integration of new information improved the baseline results. In particular, the best result was obtained at the second iteration when using bigrams. Arguably, this behaviour was due because bigrams are better suited to look for the most used collocations of an author from a small corpus. An additional experiment was carried out for both for the baseline and for the iterations performed, considering only the atrobits with information gain (IG) greater than zero (those

Table 2
Accuracy percentage after the training corpus enrichment

| | 1-gram | | Vocabulary | 2-gram | | 3-gram | |
|---|---|---|---|---|---|---|---|
| | BAYES | SVM | | BAYES | SVM | BAYES | SVM |
| BASELINE | 78.9 | 66.2 | 8377 | | | | |
| BASELINE + IG | 56.3 | 43.7 | 151 | | | | |
| E1 | 77.5 | 64.8 | 8732 | 78.9 | 66.2 | 74.6 | 64.8 |
| E1 + IG | 57.7 | 49.3 | 156 | 64.8 | 53.5 | 64.8 | 53.5 |
| E2 | 80.3 | 64.8 | 9019 | 82.9 | 74.6 | 78.9 | 66.2 |
| E2 + IG | 53.5 | 53.5 | 159 | 64.8 | 57.7 | 66.2 | 45.1 |
| E3 | 78.9 | 64.8 | 9319 | 80.3 | 66.2 | 80.3 | 64.8 |
| E3 + IG | 56.3 | 47.9 | 161 | 74.6 | 53.5 | 57.7 | 47.9 |
| E4 | 78.9 | 64.8 | 9676 | 80.3 | 66.2 | 80.3 | 64.8 |
| E4 + IG | 53.5 | 45.1 | 163 | 64.8 | 57.7 | 57.7 | 47.9 |
| E5 | 74.7 | 66.2 | 9915 | 78.9 | 68.3 | 78.9 | 68.3 |
| E5 + IG | 53.5 | 45.1 | 149 | 64.8 | 57.7 | 57.7 | 47.9 |

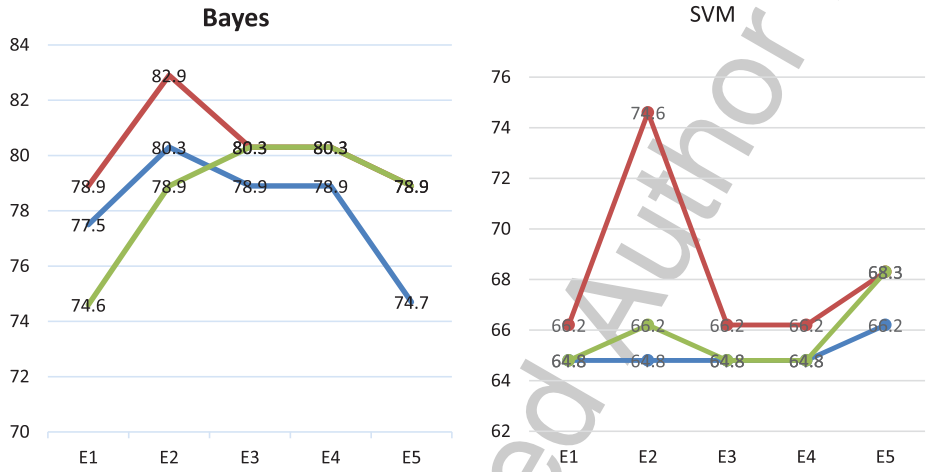


Fig. 2. Left Bayes, Right SVM (blue:1-gram, red:2-gram, green:3-gram).

attributes that serve to distinguish between classes, in thematic classification), the results are shown in Table 2.

Figure 2 show the results obtained using as classification attributes unigrams, bigrams, and trigrams using Bayes and SVM respectively for each of the iterations performed using the proposed method. As you can see, Bayes allows to have a better result of classification using the corpus of poets. With both Bayes and SVM the best result is obtained in the second iteration.

From Fig. 2, we can observe that, starting from baseline, the accuracy improves in iterations 1 and 2 and then decreases (with Bayes and SVM) this is because the incorporation of unlabeled information from the web is too much compared to the size of the original training corpus and this generates a bias in the training sets with respect to the original class. That is, 2400 snippets are downloaded and from there 60 are selected that will be incorporated into the training set in each iteration (In 5 iterations, 12,000 additional unlabeled examples are downloaded). In iteration number 3, 180 unlabeled instances have been incorporated into the training set which represents more than 50% of the size of the original corpus, that´s, as of this iteration the amount of unlabeled information coming from the web is more than the information that was originally available to decide which poet wrote a particular poem.

Figures 3 and 4 s show the results obtained by each iteration and by poet for Bayes and SVM respectively. The baseline is also shown, in order to see the impact of incorporating new information in each iteration.

In this figures we can see that the incorporation of unlabeled information helps to improve the accuracy of classification in the case of Bayes an improvement

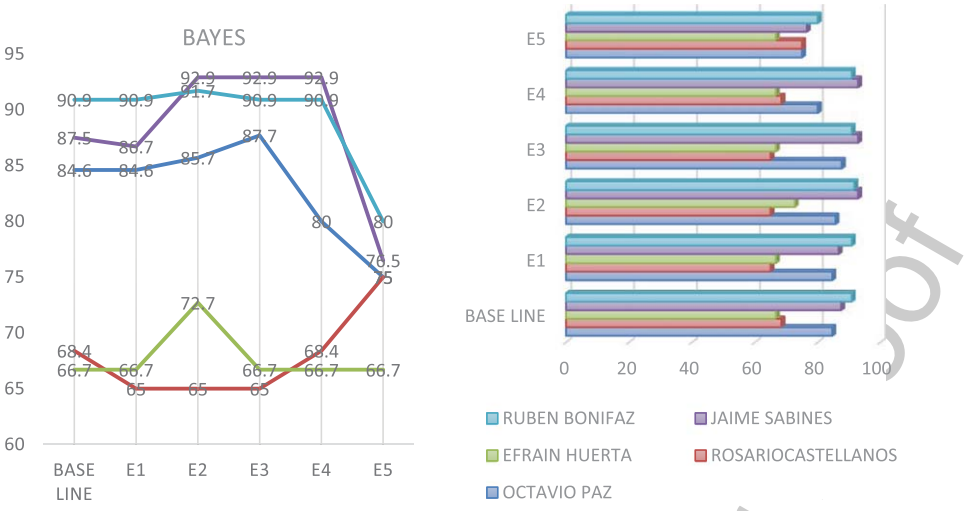


Fig. 3. Results obtained for the attribution of authorship using Bayes.

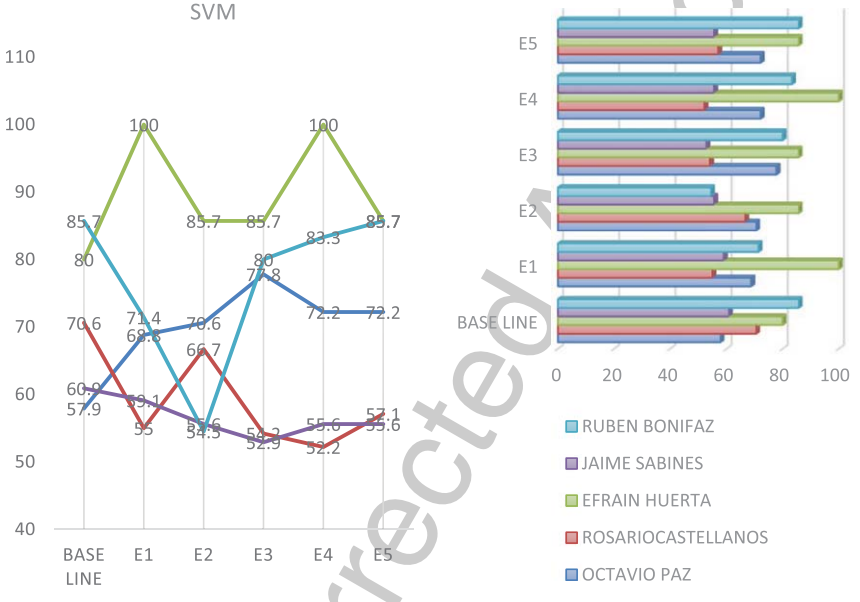


Fig. 4. Results obtained for the attribution of authorship using SVM.

is achieved for all classes, while in the case of SVM the incorporation of unlabeled information from of the web is harmful in most classes.

In spite of being preliminary results, it is surprising to verify that it is feasible to extract useful examples from the Web for the task of authorship attribution. Our intuition suggested the opposite: given that poems tend to use rare and improper word combinations, the Web seemed not to be an adequate source of relevant information for this task. That is, each author has preference topics and this information facilitates the poem classification.

## 5. Conclusions and future work

The proposed method for authorship attribution, which uses n-gram features and a semi-supervised learning approach, could outperform most common approaches for authorship attribution. Furthermore, my method, contrary to other approaches, is not too sensitive to the size of the texts and the collection, and avoids using any sophisticated linguistic analysis of documents.

The proper identification of an author, even from a poem, must consider both stylometric and topic

features of documents. Therefore, it is clear that n grams can be used as classification attributes.

Finally, from the results obtained and shown in Table 4, it can be seen that the selection of attributes that have information gain greater than zero (IG> 0), this is those attributes that help us distinguish one class from another in the thematic classification, does not help identify the writing style of poets.

## References


[1] A. Kjersti and E. y Line, Text Categorization: A survey, *Norwegian Computing Center*, (1999).

[2] F. Sebastiani, A Tutorial on Automated Text Categorization, *Istituto di elaborazione dell' Informazione*, (1999).

[3] A. Molina, Desambiguación en procesamiento del Lenguaje Natural Mediante Tecnicas de Aprendizaje Automático, *Tesis Doctoral, Dep. Sistemas Informáticos y Computación, Universidad Politécnica de Valencia*, (2004).

[4] G. Petasis, A. Cucchiarelli, P. Velardi, G. Paliouras, V. Karkaletsis and D. y Spyropoulos, Automatic Adaptation of Proper Noun Dictionaries Through Cooperation of Machine Learning and Probabilistic Methods, *Proceedings of 23 annual international ACM SIGIR conference on Research and Development in Information Retrieval*, ACM press, (2000).

[5] H. Schutze, D. Hull and J. y Pedersen, A comparison of Classifiers and Document Representations for the Routing Problem, (1995).

[6] C. Chaski, Who's at the Keyword? Authorship Attribution in Digital Evidence Investigations, *International Journal of Digital Evidence* **4**(1) (2005).

[7] A. Kaster, S. Siersdorfer and G. Weikum, Combining Text and Linguistic Document Representations for Authorship Attribution, *Workshop Stylistic Analysis of Text for Information Access*, 28th Int. SIGIR 1. MPI, Saarbrücken **2005** (2005), 27–35.

[8] M.H. Halstead, Elements of Software Science (Operating and Programming Systems Series), *Elsevier Science Inc.*, New York, NY, USA. (1977).

[9] T.J. McCabe, A complexity measure, *EEE Transaction on Software Engineering* **2**(4) (1976), 308–320.

[10] M.J. Wise, YAP3: Improved detection of similarities in computer program and other texts, *Proceedings of the Twenty-seventh SIGCSE Technical Symposium on Computer Science Education, SIGCSE ' 96, ACM*, New York, NY, USA, (1996), pp. 130–134.

[11] D. Gitchell and N. Tran, Sim: A utility for detecting similarity in computer programs, *SIGCSE Bull* **31**(1) (1999), 266–270.

[12] G. Cosma, An approach to source-code plagiarism detection and investigation using latent semantic analysis, *IEEE Transactions on Computers* **61**(3) (2008), 379–394.

[13] G. Sidorov, M.I. Romero, I. Markov, R. Guzman-Cabrera, L. Chanona-Hernandez and F. Velasquez, Deteccion automatica de similitud entre programas del lenguaje de programacion Karel basada en tecnicas de procesamiento de lenguaje natural, *Computación y Sistemas* **20**(2) (2016), 279–288. doi: 10.13053/CyS-20-2-2369

[14] S. Hsu and S. Lin, A block-structured model for source code retrieval, *Proceedings of Intelligent Information and Database Systems: Third International Conference*, ACIIDS 2011, Springer Berlin Heidelberg, Berlin, Heidelberg, (2011), 161–170.

[15] S. Schleimer, D.S. Wilkerson and A. Aiken, Winnowing: Local algorithms for document fingerprinting, *Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data, SIGMOD '03*, ACM, New York, NY, USA, (2003), pp. 76–85.

[16] G. Sidorov, A. Gelbukh, H. Gomez-Adorno and D. Pinto, Soft similarity and soft cosine measure: Similarity of features in vector space model, *Computacion y Sistemas* **18**(3) (2014), 491–504.

[17] J.P. Posadas-Duran, I. Markov, H. Gomez-Adorno, G. Sidorov, I. Batyrshin, A. Gelbukh and O. Pichardo-Lagunas, Syntactic n-grams as features for the author profiling task, *Working Notes Papers of the CLEF 2015 Evaluation Labs*, volume **1391** of CLEF '15, CEUR. (2015).

[18] H. Gomez-Adorno, G. Sidorov, D. Pinto and I. Markov, A graph based authorship identification approach, *Working Notes Papers of the CLEF 2015 Evaluation Labs*, volume **1391** of CLEF '15, CEUR. (2015).

[19] G. Sidorov, H. Gomez-Adorno, I. Markov, D. Pinto and N. Loya, Computing text similarity using tree edit distance, *Proceedings of the Fuzzy Information Processing Society (NAFIPS) held jointly with 2015 5 World Conference on Soft Computing (WConSC)*, 2015 Annual Conference of the North American, NAFIPS '15, IEEE, pp. 1–4.th (2015).

[20] G. Sidorov, Construccion no lineal de n-gramas en la linguistica computacional: n-gramas sintacticos, filtrados y generalizados. Mexico. (2013).

[21] R. Guzmán-Cabrera, M. Montes-y-Gómez, P. Rosso and L. Villaseñor-Pineda, Using the Web as corpus for self-training text categorization, *Inf Retrieval* **12** (2009), 400–415. DOI 10.1007/s10791-008-9083-7