Proceedings of the Second European IAFL Conference on Forensic Linguistics / Language and the Law, Barcelona 2006.

An Extremely Simple Authorship Attribution System

Rogelio Nazar
rogelio.nazar@upf.edu

Marta Sánchez Pol
marta.sanchez@upf.edu

Institut Universitari de Lingüística Aplicada
Universitat Pompeu Fabra
Pl. de la Mercè 10-12
08002 Barcelona

Abstract: In this paper we present a very simple yet effective algorithm for authorship attribution. By this term we mean the act of telling whether a certain text was or was not written by a certain author. We shall not discuss the advantages or applications of this activity, but we propose a method for doing it in an automatic and instantaneous way, neither considering the language of the texts nor undertaking any kind of text preprocessing like tokenization or part of speech tagging. We have conducted an experiment that shows how authorship attribution can be seen as a text categorization problem. That is to say, each author represents a category and the documents are the elements to be classified. Text categorization has became a very popular issue in computational linguistics and it has developed to great complexity, motivating a huge amount of literature. However, in this article we show a basic method applied to authorship attribution. This program is language independent because it uses purely mathematical knowledge: an n-gram model of texts. It works in a very simple way and is therefore easy to modify. In spite of its simplicity, this program is capable of classifying documents by author obtaining more than 90% of accuracy. As an example, we present an experiment carried out with a rather homogeneous corpus composed of 100 short newspaper articles (around 400 words each) from 5 different authors. As a final comment, it is worth saying that this system can be used as a general purpose document classifier, for example by content instead of authorship, because it only reproduces the criterion that it learned during the training phase.

Keywords: text categorization by authorship; quantitative stylistics; n-grams analysis; language-independent text categorization.

1 *Introduction*

We present here a method for authorship attribution, meaning an algorithm to determine if a person is the author of a text. No special knowledge from the reader is required to understand this paper, but with a basic background of any programming language it should not take more than a few hours to implement it.

Originally this design was conceived for the classification of documents by topic. In some preliminary experiments using IULA's Technical Corpus results were around %90 of accuracy in categories like economics, computer sciences, medicine, etc. But the technique resulted also appropriate for the categorization by author. This versatility is explained by its condition of a supervised learning algorithm. It can be trained to solve different problems, if these are in fact classification problems. In this kind of algorithms the execution is divided in a learning phase and a test phase. During the first phase a user is supposed to train the system with examples, assigning a set of objects to different categories. Once this training phase is completed, the test phase will tell us if the algorithm is capable of classifying new objects correctly.

Text categorization has been indeed the canonical point of view of statistical methods when dealing with authorship problems since the sixties (Mosteller and Wallace, 1984) and continues to be in more recent advances in the area (Stamatatos et al. 2000). In a problem like this we try to determine the authorship of a text by comparing its properties with those of the texts whose authorship is known. This paper goes along the same line, but respects the principle of Occam's Razor, as frequently recommended by Mitchell (1997). This is, if one has to choose between two theories or models that can both acceptably explain some data, the most simple one is always preferable.

Nevertheless, the simplicity ideal does not seem to be important in the context of machine learning, where algorithms usually need big amounts of training data. Generally speaking, in statistical approaches, the higher the numbers, the better the results. It is common to see experimenters that require thousands of documents for training and test, like those reported by Manning and Schütze (1999), for example Entropy Maximization that achieves 88.6% of accuracy after training with 9,603 documents or Perceptrons, showing 83% accuracy after the same training.

This is quite impractical in forensic linguistics because frequently this amount of data is not available in case of an authorship dispute. On the contrary, usually few documents are available as a training set and most of the time just one document as a test set.

Simplicity of the approaches is desirable too when dealing with linguistic data, although it is not always possible. Because of the variety of linguistic phenomena, we would prefer a statistical approach rather than a system based on rules of the type “if-then”. Using these rules it is possible to build decision trees that can be interpretable by humans, but that can also grow to great complexity. This is the so called symbolic approach (Sebastiani, 2002) and its major drawback is the cost of manually building a set of rules capable of capturing the diversity of occurrences in real life texts.

The use of this type of rules makes possible the classification of texts by the presence or absence of certain markers. However, in some cases we would like to quantify this presence, that means to associate a real value to a marker instead of just a binary condition. When this is the case, statistical measures outperform symbolic systems. Furthermore, they can capture unexpected phenomena. With statistical approaches it is even possible to classify objects without training data. This would be unsupervised learning, like document clustering. This is a technique used to classify objects without knowing what or how many categories will be seen in the test set. In our approach here we use, however, the supervised type of learning.

2 The algorithm

Known and unknown texts are represented as vectors of term weights. We may understand a vector as a set of attributes and values, regardless of their order or sequence. Vectors of known texts of the same author are summed together into the same class. The elements of these vectors, the terms, are bigrams associated to their frequency of occurrence, being a bigram in this context a sequence of two words as they appear in the text. A word is defined as a sequence of characters between spaces and the weight is just its frequency of occurrence after the corpus has been normalized. Here normalization means that all classes in the training set have the same extension, so text in the training set is discarded until this condition is met. The only preprocess we undertake is the separation of punctuation marks from the words, thus considering them tokens on their own. Table 1 shows some examples of Spanish bigrams.

Features	vector 1	vector 2	...
no me	5	0	...
, que	4	6	...
el hombre	3	2	...
. y	3	4	...
a su	3	4	...
su señora	2	0	...
, como	2	3	...
me he	2	3	...
...	...	...	...

Table 1: Fragment of a matrix of Spanish bigrams.

In the Table 1, a vector could be a text to classify and the other vector could represent one of the classes. There can be n vectors and d features. During the test phase the disputed text-vector is compared to each of the vectors that represents an author. To do the comparison we use a similarity measure that consists of the sum of the values of the features of both vectors, choosing the pair of vectors (disputed text and author) that results with the highest number.

$$\max_{1 \leq j \leq n} \sum_{i=1}^{d} (x_i^0 + x_i^j)$$

This similarity measure could be interpreted as a variant of the Matching Coefficient, that gives a higher score to the pair of vectors that has more components in common, without any account of the values associated to the components. Our measure, instead, is applicable to real values instead of binary ones.

We register every bigram in the corpus, therefore there is no need for feature selection. There is no weighting of terms neither, although this is usually done in Information Retrieval (Sebastiani, 2002). Some may consider only terms having a low or medium document frequency because they are the most informative ones. A word like “the”, for example, appears in almost every English text and therefore it is not very informative. Another common technique is the removal of words of the training set that have a frequency under a certain threshold, for example deleting all hapax legomena or dislegomena. We did not try anything like this, nor the removal of function words (such as articles, prepositions, conjunctions, etc.); neither stemming or lemmatization. We also renounced to perform the extraction of specific style markers; the sentence and/or chunk boundaries detection; the account of common word frequencies; measures of vocabulary richness; sentence length; etc. (Sánchez Pol, 2006; Stamatatos, 2000) as well as syntactic annotation or part of speech tagging, excluding therefore the study of the sequences of these tags (Spassova, 2006).

There is of course no additional or exogenous knowledge, as pointed out by Sebastiani (2002), meaning metadata or data provided by an external source. The classification is supposed to be done only on the basis of endogenous knowledge, the information extracted from the documents.

Another arbitrary decision has been to consider all categories as non-overlapping, which means that we do not consider the possibility of having collaborative writing of documents, although this could possibly be the case and in fact is quite frequent, specially in scientific domains.

This learning method presents some similarity with the one proposed by Peng et. al (2003) because they also use n-gram models of texts, but in their case at the character level. They claim to have obtained, with no feature selection or pre-processing, superior results than more sophisticated learning techniques. The character level is a smart choice because it reduces the sparse data problems of vocabulary models. N-grams of characters are much more frequent than n-grams of words and at the same time in the character sequences there is less diversity.

However, in the experiments we conducted so far at the word level n-grams we did not encounter problems of vocabulary size, even when we did not use special hardware. A probable explanation could be that we are not using very big corpora. The maximum amount of data tested at the moment was a categorization by topic (three categories) using a corpus of 500 documents and a total extension of approximately 6.5 million words. This took 34 minutes[1] with a SUN Enterprise 250 machine (2 x UltraSPARCII a 400MHz, 2Gb Ram). The corpus of the authorship experiment we show in this paper is much smaller. It has a total extension of around 40,000 words. It is the same that Sánchez Pol (2006) used in her master thesis: a hundred short newspaper articles (the think piece type) written in Spanish by five different authors, 20 articles each, having all almost the same extension (400 words). It takes 7 seconds for the algorithm to execute on this corpus.

3 Results

Figure 1 shows the learning curve where we can appreciate how precision (vertical axis) increases as more percentage of the corpus is used for training (horizontal axis). We see that it reaches 92% when we use half of the corpus (50 documents) for training and the others for test.

In these results we consider categorization performance by the overall accuracy, which is the total number of correctly classified texts divided by the total number of texts. We do not

[1] This comparison serves only as a rough reference because part of this time is consumed during the extraction of the corpus from IULA's server (http://www.bwananet.iula.upf.edu), while in the experiment shown in this paper the texts were already available, so this step was unnecessary.

consider recall because we force it to be %100 by not letting any document without category assignment.

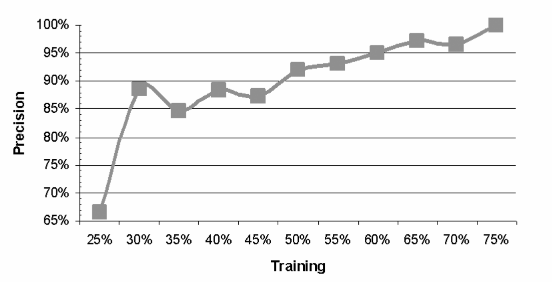

Figure 1: The learning curve of the algorithm

These results may be comparable with the 90% accuracy reported by Peng et al. (2003) and the 72% accuracy in Stamatatos et al. (2000). One caveat is that they used a different corpus: 200 texts written by 10 authors using 100 for training and the rest for test, while we used 100 texts written by 5 authors..

4 Statistical Significance

One way to determine the statistical significance of this experiment is to conduct Pearson's Chi-square Test. In this way we can determine the probability of true of a null hypothesis, that in this case would be that the documents have been classified in a random manner so we had this outcome by chance. If this was true, and considering that we have 5 different authors for each document to assign and that we make 50 trials, then we would expect a quantity of successes as a random variable with a uniform probability distribution. That is, we would expect a mean of 10 successes, while the observed data is that we have 46 successes.

The chi-square helps us to compare the expected mean with the observed values, and then to determine how likely it is that the sample belongs to the uniform distribution of the null hypothesis.

$$\chi^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i}$$

With our numbers this yields a value of 162, and with 4 degrees of freedom (number of authors minus 1) this corresponds to a p < 0.00001.

Another way to do the same test is to plot the probability distribution that we would have if the null hypothesis was true. Figure 2 shows that we would have an expected mean of 10 successes in a sample of 50 trials. If we had a sample with 5 or 15 successes, or any other value under the curve, then this outcome could perfectly be due to chance. But the sample of 46 successes is very much out of this interval.

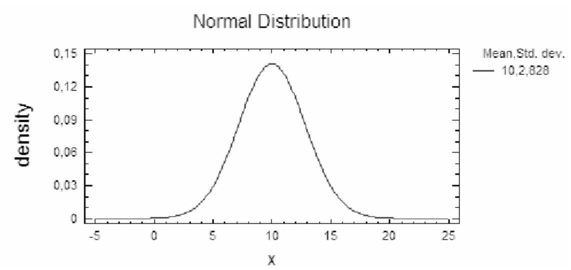

Figure 2: Probability distribution for the expected mean.

5 Conclusions

If the simplicity ideal is still relevant for computer science and linguistics, then we can be satisfied with the achievement of an original system that follows the “think small” philosophy, a toy algorithm we may say, specially if we compare it to other very complex machine learning algorithms that are available today in the market.

Finally we will insist on the idea that this is a general purpose classifier. It reproduces the criteria in which it was trained, so it can perform classifications not only by author but, for example, by language, by topic, by degree of specialization, etc.

6 References

Manning, C.; Schütze, H. (1999). Foundations of Statistical Natural Language Processing. Cambridge. MIT Press.

Mitchell, T. (1996). Machine Learning. New York. McGraw Hill.

Mosteller, F.; Wallace, D. (1984). Applied Bayesian and classical inference the case of the Federalist papers. New York. Springer.

Peng, F.; Schuurmans, D.; Keselj, V; Wang, S. (2003) “Language Independent Authorship Attribution using Character Level Language Models”. Proceedings, 10th Conference of the European Chapter of the Association for Computational Linguistics, Budapest, 267--274

Sánchez Pol, M. (2006). Proposta d'un mètode d'estilística per a la verificació d'autoria: els límits de l'estil idiolectal. Barcelona, Institut Universitari de Lingüística Aplicada. [Master's thesis supervised by Dra. M.Teresa Turell.]

Sebastiani, F. (2002), "Machine learning in automated text categorization". ACM Computing Surveys (CSUR). Volume 34 Issue 1. ACM Press.

Spassova, M. (2006), Las marcas sintácticas de atribución forense de autoría de textos escritos en español. Barcelona, Institut Universitari de Lingüística Aplicada. [Master's thesis supervised by Dra. M.Teresa Turell.]

Stamatatos, E.; Kokkinakis, G.; Fakotakis, N. (2000), "Automatic text categorization in terms of genre and author", Computational Linguistics, Volume 26 Issue 4, MIT Press.