Using Bag-of-words to Distinguish Similar Languages: How Efficient are They?

Marcos Zampieri
Saarland University
Uni Campus Nord - Building A2.2
D-66123, Saarbrücken, Germany
Email: marcos.zampieri@uni-saarland.de

***Abstract*—This paper presents a number of experiments describing the use of machine learning algorithms and bag-of-words to the task of automatic language identification. The paper focuses on the identification of language varieties, which is a known weakness of general purpose language identification methods. This question was addressed by a number of studies in the recent years, most of them relying on character n-gram language models. In this paper, I experiment simple bag-of-words and compare the results with previously proposed n-gram-based approaches. To perform these classification experiments three algorithms were used: Multinomial Naive Bayes (MNB), Support Vector Machines (SVM) and the J48 classifier.**

I. Introduction

There are a number of situations in which the source language of a document is unknown. Computational methods can therefore be applied to automatically detect a text's source language before carrying out tasks such as machine translation or information retrieval. This task is known as automatic language identification or simply language identification. The distinction between languages is often based on n-gram-based language models, calculated at word or at the character level.

Language identification is a well-established research topic and its origins can be traced back to the work of Ingle [1]. State-of-the-art general purpose language identification methods achieve performance usually over 95% accuracy such as in Lui and Baldwin (2012) [2] and Brown (2013) [3]. Even so, there are two known bottlenecks in this task. The first of them is language identification in very short often 'noisy' pieces of text [4] which contain multilingual, code-switching and non-standard features.

The second bottleneck is the identification of similar languages and (when necessary) the distinction between different varieties or dialects. Serbian and Croatian or Swedish and Danish are examples of closely related languages which share a great deal of lexical and grammatical features making it difficult for algorithms to distinguish them automatically. Identifying the language variety of a text written in English, Portuguese or French is even more challenging and only recently has this aspect of language identification received more attention.

Identifying closely related languages and varieties will be addressed in this paper using machine learning algorithms. As evidenced in section II-A, recently some studies have been published attempting to distinguish between closely related languages and varieties [5], [6], [7].

A. Language Identification: A Classification Task

Language identification is essentially a document classification task that consists of assigning documents to classes or categories that are represented by a finite set of labels. The type of classification used for language identification is single-label classification, allowing one label to be attributed to each instance (text) and represented by the following function:

$$f_{class} : \chi \rightarrow \lambda \tag{1}$$

In 1, χ is the sample space and λ is a set of class labels. The classification function then maps the relation between a label $y \in \lambda$ to all instances of a given dataset. In language identification, the labels $y \in \lambda$ are a set of languages that the method tries to attribute to each text.

To the best of my knowledge, the vast majority of approaches developed to identify similar languages, rely on character n-gram language models. Only a few approaches, such as the case of Huang and Lee [6], use bag-of-words (BoW) to solve this problem. BoW are a simple form of data representation widely used in text categorization and information retrieval, but not yet exhaustively explored for language identification. This work aims to fill this gap comparing their performance with n-gram-based methods.

II. Related Work

Ingle [1] was one of the first studies to be published on language identification. He applied Zipf's law distribution to order the frequency of short words in text and used this information for language identification. The studies published by Beesley [8] and later by Dunning [9] introduced the use of character n-gram language models to the task, which are still the basis of most state-of-the-art methods. Dunning [9] reports over 99% accuracy in distinguishing English and Spanish texts. In this approach, the likelihood of character n-grams is calculated using Markov models.

After Dunning, several studies using n-gram language models were published, among them is the work of Cavnar and Trenkle [10]. This study uses a list of the most frequent character n-grams in a corpus. A metric establishes n-gram profiles and calculates a simple rank-order statistics that determines

how far out of place an n-gram in one profile is from its place in the category as figure 1 shows.

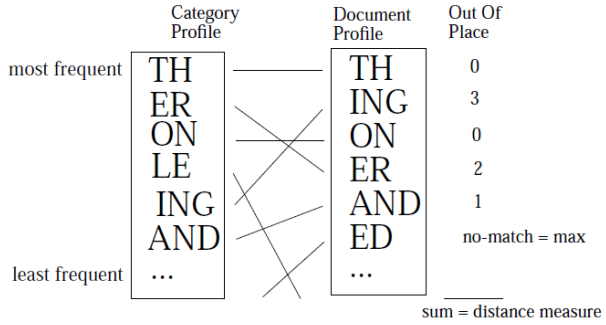

Fig. 1. Out-of-place Measure (Cavnar and Trenkle, 1994)

Based on the authors' description [10], the n-gram 'ING' is at rank 2 in the document, but at rank 5 in the category, therefore 3 ranks out of place. If an n-gram (e.g. 'ED') is not in the category profile, it takes a maximum out-of-place value (arbitrarily defined). The sum of all of the out-of-place values for all n-grams is the distance measure for the document from the category. The algorithm then applies what they called a 'Find Minimum Distance' function. This function takes the distance measures from all of the category profiles to the document profile, and picks the smallest one.

Grefenstette [11] compares two methods of language identification: a trigram approach inspired by the work of Beesley [8] and Cavnar and Trenkle [10] and the frequent short word approach proposed by Ingle [1]. Grafenstette points out the simplicity of both methods and the advantage of character-based approaches when dealing with texts shorter than 15 words. According to this study, shorter sentences are often titles and section headings, which might not contain any of the short words used for classification in Ingle's approach.

A couple of other comparative studies are worth mentioning: Vojtek and Belikova [12] compare two methods based on Markov processes including the aforementioned method proposed by Dunning [9]. Padró and Padró [13] compared the performance of three language identification methods: Markov models, trigram frequency vectors and n-gram based text categorization [10] and finally, Groethe et al. [14] compare methods based on three features: short words, frequent words and character n-grams.

Machine learning techniques have been used in language identification over the years. [15] proposed the use of machine learning as an alternative to Markov-based approaches. [16] applies a centroid-based classification approach, widely used in text classification. Although most language identification studies involve supervised learning strategies, there were a couple of attempts to perform the task by using unsupervised methods such as [17]. In this study, authors propose a hybrid method for language identification that includes k-means clustering.

The Internet is an interesting application for language identification as documents available on the Internet are often unidentified regarding source language. The same document may contain more than one language as well as non-standard spelling. A substantial amount of user-generated content is considered to be 'noisy' and often too short which makes it difficult for computer programs to process them. This originates one of the previously mentioned bottlenecks in language identification.

A couple of language identification methods developed for internet date include [18], [19] and the *LIGA* algorithm [4] and [20] developed for short internet texts such as *tweets*. More recently, Nguyen and Dogruoz [21] propose a language identification method at word level to distinguish between Dutch and Turkish in computer-mediated communication. For evaluation the method, the authors use a large online forum for Turkish-Dutch speakers living in the Netherlands.

Among the most recent language identification studies is a tool called *langid.py* developed by Lui and Baldwin (2012) [2]. *langid.py* is an off-the-shelf general purpose language identification tool which achieved results of up to 94.7% accuracy, outperforming similar tools such as *TextCat* [10] and *GoogleAPI*. Lui and Baldwin's approach uses a multinomial Naive Bayes classifier and information gain (IG) for feature selection [22]. The method was tested using a dataset containing 97 languages and 5 different domains and the authors observed not only performance superior to similar tools but also processing speed.

To my best knowledge, the most recent general-purpose study on language identification is the one by Brown (2013) [3]. This language identification method uses cosine similarity on a filtered and weighted subset of the most frequent n-grams with optional smoothing. The software, called *whatlang* was applied to a collection of documents written in 1,100 languages in which each document contained at most 65 characters. The performance of Brown's algorithm reached 99.2% accuracy using smoothing and 98.2% without smoothing.

A. Distinguishing Similar Languages and Varieties

Distinguishing similar languages is one of the aforementioned bottlenecks of language identification and this aspect has been receiving more attention in the past few years. Ljubešić et al. [5] proposed a computational model for the identification of Croatian texts in comparison to Slovene and Serbian. The study reports 99% recall and precision in three processing stages. One of these processing stages, includes a 'black list', a list of words that appear only in Croatian texts, making the algorithm perform better. Tiedemann and Ljubešić [23] improve this method and apply it to Bosnian, Serbian and Croatian texts. The study reports significantly higher performance than general purpose language identification methods, such as *TextCat* and *langid.py*.

Ranaivo-malancon [24] presents a semi-supervised character-based model to distinguish between Indonesian and Malay, two closely related languages from the Austronesian family. The study uses the frequency and rank of character trigrams derived from the most frequent words in each language, lists of exclusive words, and the format of numbers

(Malay uses decimal point whereas Indonesian uses comma). The author compares the performance of this method with the performance obtained by *TextCat*.

The methods applied to language varieties and dialects are similar to those applied to similar languages[1]. One of the methods proposed to identify language varieties is the one by Huang and Lee [6]. This study presented a bag-of-words approach to classify Chinese texts from the mainland and Taiwan with results of up to 92% accuracy.

Another study is the one published by Zampieri and Gebre [27] for Portuguese. In this study, the authors proposed a log-likelihood estimation method along with Laplace smoothing to identify two varieties of Portuguese (Brazilian and European). Their approach was trained and tested in a binary setting using journalistic texts with accuracy results above 99.5% for character n-grams. The algorithm was later adapted to classify Spanish texts using not only the classical word and character n-grams but also POS distribution [28].

The experiment described by Mohkov [29] take into account not only French language varieties but also a temporal dimension. This system was one of the six systems to participate in the DEFT2010[2] shared task held in Montreal. In this evaluation campaign, systems aimed to classify French journalistic texts with respect to their geographical location as well as the decade in which they were published.

B. Using Bag of Words

Bag-of-words have been widely used in text categorization problems. They are a very simple way of representing data that assumes no independence between words. In bag-of-words, texts (instances to be classified) are represented by a word vector with an n number of entries. These n entries correspond to all words found in the corpus and catalogued in a dictionary. The entries n receive a number y depending on the presence or absence of n in the instance.

As previously mentioned, apart from the study published by Huang and Lee [6] very few has been said about the use of bag of words for language identification. This is basically because language identification methods developed to distinguish similar language or language varieties use the same methods applied to general purpose language identification. In this paper, I will compare the performance of the system using machine learning algorithms and bag-of-words as features to the methods that use n-gram language models described in [7]. I compared the BoW results firstly to to the best n-gram language models and subsequently to the results obtained using word unigram models.

III. Methods

For these experiments a number of comparable journalistic corpora were compiled. The number of texts sampled were equivalent to previous datasets used in [27] and [7] to allow for comparison. The languages, source and year of publication of the texts are presented next:

Language	Code	Corpora	Year
Argentinan Spanish	ARG	La Nacion	2008
Brazilian Portuguese	BRA	Folha de São Paulo	2004
Hexagonal French	FRA	Le Monde	2008
European Portuguese	POR	Diario de Noticias	2008
Quebecian France	QUE	Le Devoir	2008
Peninsular Spanish	SPA	El Mundo and El Pais	2008

TABLE I
CORPORA

The corpora presented above were arranged in bag-of-words and used as features to feed machine learning classifiers. The three classifiers used are available in the WEKA Machine Learning Workbench [30]. For this pre-processing step Python scripts were used to arrange files in the WEKA ARFF format.

A. Algorithms

The three machine learning algorithms used for this study are Multinomial Naive Bayes (MNB), Support Vector Machines (SVM) and J48. These algorithms are widely used in NLP and text classification and they differ substantially in the way they perform classification. For this paper, standard distributions of these three algorithms available in the WEKA package were used and no parameter was changed. A short overview of each classifier based on what is described by Witten and Frank [30] is presented as follows:

1) Multinomial Naive Bayes: Multinomial Naive Bayes (MNB) as the name suggests is based on Bayes theory and probability represented by the following equation:

$$P(A|B) = \frac{P(A|B)P(A)}{P(B)} \tag{2}$$

As described in [31], MNB applied to text classification computes class probabilities for a given document and the set of classes is represented by C. MNB assigns a text document t_i to the class with the highest probability $P(c|t_i)$ given by the equation below for $c \in C$:

$$P(c|t_i) = \frac{P(t_i|c)P(c)}{P(t_i)} \tag{3}$$

Broadly speaking, Naive Bayes classifiers work under the assumption that the presence or absence of a particular feature of a class is not related to the presence or absence of any other feature, also sometimes referred as a Markov assumption. This independence assumption makes Naive Bayes classifiers particularly useful for supervised learning and makes them extremely fast when compared to other learning algorithms.

Kibriya et al. [31] discuss the use of MNB and the transformation steps that lead to 'transformed weight-normalized complement naive Bayes' (TWCNB) to the task of text classification applied to four datasets. Researchers observed that

[1]It is beyond the scope of this paper to discuss the fine line between languages, varieties and dialects. More to this discussion can be found in Clyne [25] and Chamber and Trudgill [26].

[2]http://www.groupes.polymtl.ca/taln2010/deft.php (in French)

MNB and TWCNB obtained better performance when applied to TF-IDF frequency data instead of BoW. In the present papers, we are not exploring the influence of TF-IDF index such as in [32] and leaving this aspect for future experiments.

2) Support Vector Machines (SVM): Support Vector Machines (SVM) are non-probabilistic binary classifiers. Given a set of instances, each of them belonging to one of two categories, SVM classifiers build models that assign new examples to each of the classes. An SVM model can be represented and understood as points in space. These points are mapped and the points belonging to the two categories are usually as wide as possible to determine classification. The implementation available in WEKA and used in this work is the one by Platt [33], named Sequential Minimal Optimization (SMO).

3) J48: The J48 algorithm is a decision tree based algorithm which is an adaptation of the popular C4.5 classifier developed by Quinlan [34]. C4.5 is an extension of the ID3 algorithm developed by the same author. C4.5 builds decision trees using the concept of information entropy (a measure of uncertainty of a random variable). As most decision tree classifiers, in these experiments the J48 algorithm was significantly slower than the other two classifiers.

IV. Results

This section presents the results obtained when classifying language varieties: European and Brazilian Portuguese; Argentinian, Mexican, Peruvian and Peninsular Spanish and Hexagonal and Quebec French.

For these experiments 1,000 documents were used. They were split in 2 partitions of 500 documents each, one for training and one for testing. This amount of data was used to compare the performance of the machine learning methods to discriminative method described in Zampieri and Gebre [27].

Results ranged from 0.988 for Portuguese using MNB and 0.865 for Spanish using the J48 classifier. The best average performance was obtained by the MNB classifier with 0.968 accuracy. In order to evaluate the extent to which these methods are effective in identifying these languages, I compared the performance obtained using machine learning and BoW with the results obtained by the traditional character and n-gram approaches and the discriminative log-likelihood-based algorithm presented in [27] and [7].

The accuracy results obtained using bag-of-words and the three aforementioned algorithms are presented in II.

Language	Classes	MNB	SVM	J48
Portuguese	2	0.988	0.987	0.942
Spanish	4	0.943	0.936	0.865
French	2	0.972	0.955	0.950
Average	2.66	0.968	0.959	0.919

TABLE II
Classification Results

Table III presents the comparison between the results of this paper (all obtained using MNB) and the best results obtained with the discriminative method by Zampieri and Gebre [27].

Language	Best Result	Comparison	Feature	Difference
Portuguese	0.988	0.998	C 4-grams	- 1.0 pp
Spanish	0.943	0.876	W 2-grams	+ 6.7 pp
French	0.972	0.990	C 3-grams	- 1.8 pp

TABLE III
Comparison with N-Gram-based Methods

Results are 6.7 percentage points better for Spanish (4 classes), 1 percentage point worse for Portuguese and 1.8 percentage points worse for French. With respect to these methods it is important to mention that they use different feature sets two of them relying on characters and one on word bigrams. Portuguese, for example, has differences in orthography between their two main varieties which enable algorithms to distinguish between European and Brazilian at the character level with good accuracy. On the other hand, word bigrams take syntax into account and this is an aspect of language that bag-of-words cannot handle.

To allow for a more fair comparison between methods, next I compared the best result obtained in this paper to the best word unigram result obtained by the likelihood method [27].

Language	Best Result	Word Unigram	Difference
Portuguese	0.988	0.996	- 0.8 pp
Spanish	0.943	0.848	+ 9.5 pp
French	0.972	0.968	+ 0.4 pp

TABLE IV
Comparison to word uni-gram models

The main differences between the word unigram method in [27] and the BoW presented here are probability calculation and smoothing. In the aforementioned method, authors use Laplace probability distribution with add one smoothing for unseen words. Bag-of-words are simpler than the unigram models and do not add any value for unseen tokens.

V. Conclusion and Future Work

This paper presented language identification experiments using machine learning classifiers and BoW focusing on language varieties. With exception of [6], BoW were not substantially explored in language identification and this work fills this gap. Results show that the method has performance comparable to state-of-the-art methods based on n-gram language models with small loss of performance in some of the cases.

For language varieties, the method outperforms a word unigram method in 2 out of 3 cases. As briefly mentioned, BoW methods are conceptually simpler than n-gram language models and for that reason the results obtained in these experiments are very interesting for language identification and more broadly to text classification.

I will continue to experiment new approaches to language identification in the near future. The next step is the integration of language varieties to broader identification schemes. This will provide an estimation of how good can these computational models distinguish similar languages in real-world

settings and the extent to which they can be integrated to existing NLP tools.

ACKNOWLEDGMENTS

The author would like to thank Binyam Gebrekidan Gebre and Sascha Diwersy for their help and for valuable discussions throughout this work.

REFERENCES

[1] N. Ingle, *A Language Identification Table*. Technical Translation International, 1980.

[2] M. Lui and T. Baldwin, "langid.py: An off-the-shelf language identification tool," in *Proceedings of the 50th Meeting of the ACL*, 2012.

[3] R. Brown, "Selecting and weighting n-grams to identify 1100 languages," in *Proceedings of the 16th International Conference on Text Speech and Dialogue (TSD2013), Lecture Notes in Artificial Intelligence (LNAI 8082)*. Pilsen, Czech Republic: Springer, 2013, pp. 519–526.

[4] E. Tromp and M. Pechnizkiy, "Graph-based n-gram language identification on short texts," in *Proceedings of the Twentieth Belgian Dutch Conference on Machine Learning (Benelearn 2011)*, 2012, pp. 27–34.

[5] N. Ljubešić, N. Mikelic, and D. Boras, "Language identification: How to distinguish similar languages?" in *Proceedings of the 29th International Conference on Information Technology Interfaces*, 2007.

[6] C. Huang and L. Lee, "Contrastive approach towards text source classification based on top-bag-of-word similarity," in *Proceedings of PACLIC 2008*, 2008, pp. 404–410.

[7] M. Zampieri, B. G. Gebre, and S. Diwersy, "Classifying pluricentric languages: Extending the monolingual model," in *Proceedings of the Fourth Swedish Language Technlogy Conference (SLTC2012)*, Lund, Sweden, 2012, pp. 79–80.

[8] K. Beesley, "Language identifier: A computer program for automatic natural-language identification of on-line text," in *Proceedings of the Annual Conference of the American Translators Association*, 1988, pp. 57,54.

[9] T. Dunning, "Statistical identification of language," Computing Research Lab - New Mexico State University, Tech. Rep., 1994.

[10] W. Cavnar and J. Trenkle, "N-gram-based text catogorization," *3rd Symposium on Document Analysis and Information Retrieval (SDAIR-94)*, 1994.

[11] G. Grefenstette, "Comparing two language identification schemes," in *Proceedings of JADT 1995, 3rd International Conference on Statistical Analysis of Textual Data*, Rome, 1995.

[12] P. Vojtek and M. Belikova, "Comparing language identification methods based on markov processess," in *Slovko, International Seminar on Computer Treatment of Slavic and East European Languages*, 2007.

[13] M. Padró and L. Padró, "Comparing methods for language identification," *Procesamiento del Lenguaje Natural*, no. 33, pp. 155–162, 2004.

[14] L. Groethe, E. De Luca, and A. Nürnberger, "A comparative study on language identification methods," in *Proceedings of LREC*, 2008.

[15] H. Combrinck and E. Botha, "Text-based automatic language identification," in *Proceedings of the 6th Annual South African Workshop on Pattern Recognition*, 1994.

[16] H. Takçı and T. Güngör, "A high performance centroid-based classification approach for language identification," *Pattern Recognition Letters*, vol. 3, pp. 2077–2084, 2012.

[17] A. Amine, Z. Elberrichi, and M. Simonet, "Automatic language identification: an alternative unsupervised approach using a new hybrid algorithm." *International Journal of Computer Science and Applications*, vol. 7, pp. 94–107, 2010.

[18] B. Martins and M. Silva, "Language identification in web pages," *Proceedings of the 20th ACM Symposium on Applied Computing (SAC), Document Engineering Track. Santa Fe, EUA.*, pp. 763–768, 2005.

[19] R. Rehurek and M. Kolkus, "Language identification on the web: Extending the dictionary method," in *Proceedings of CICLing. Lecture Notes in Computer Science*. Springer, 2009, pp. 357–368.

[20] J. Vogel and D. Tresner-Kirsch, "Robust language identification in short, noisy texts: Improvements to liga," in *Third International Workshop on Mining Ubiquitous and Social Environments (MUSE 2012)*, 2012.

[21] D. Nguyen and S. Dogruoz, "Word level language identification in online multilingual communication," in *Proceedings of EMNLP2013*, Seatle, USA, 2013.

[22] M. Lui and T. Baldwin, "Cross-domain feature selection for language identification," in *Proceedings of 5th International Joint Conference on Natural Language Processing*. Chiang Mai, Thailand: Asian Federation of Natural Language Processing, November 2011, pp. 553–561.

[23] J. Tiedemann and N. Ljubešić, "Efficient discrimination between closely related languages," in *Proceedings of COLING 2012*, Mumbai, India, 2012, pp. 2619–2634.

[24] B. Ranaivo-Malancon, "Automatic identification of close languages - case study: Malay and indonesian," *ECTI Transactions on Computer and Information Technology*, vol. 2, pp. 126–134, 2006.

[25] M. Clyne, *Pluricentric Languages: Different Norms in Different Nations*. CRC Press, 1992.

[26] J. Chambers and P. Trudgill, *Dialectology (2nd Edition)*. Cambridge University Press, 1998.

[27] M. Zampieri and B. G. Gebre, "Automatic identification of language varieties: The case of Portuguese," in *Proceedings of KONVENS2012*, Vienna, Austria, 2012, pp. 233–237.

[28] M. Zampieri, B. G. Gebre, and S. Diwersy, "N-gram language models and POS distribution for the identification of Spanish varieties," in *Proceedings of TALN2013*, Sable d'Olonne, France, 2013.

[29] S. Mokhov, "A marf approach to deft2010," in *Proceedings of TALN2010*, Montreal, Canada, 2010.

[30] I. Witten and E. Frank, *Data mining: practical machine learning tools and techniques*. Morgan Kaufmann Publishers, 2005.

[31] A. Kibriya, E. Frank, B. Pfahringer, and G. Holmes, "Multinomial naive bayes for text categorization revisited," in *Proceedings of the Australian Conference on Artificial Intelligence*, 2004, pp. 488–499.

[32] B. G. Gebre, M. Zampieri, P. Wittenburg, and T. Heskens, "Improving native language identification with tf-idf weighting," in *Proceedings of the 8th NAACL Workshop on Innovative Use of NLP for Building Educational Applications (BEA8)*, Atlanta, USA, 2013.

[33] J. Platt, "Fast training of support vector machines using sequential minimal optimization," in *Advances in Kernel Methods Support Vector Learning*, B. C. Schoelkopf, B. and A. Smola, Eds., 1998.

[34] J. R. Quinlan, *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, San Mateo, CA, 1993.