Comparing Ranking-based and Naive Bayes Approaches to Language Detection on Tweets*

Comparando el enfoque basado en rankings con el bayesiano para la detección de idioma en tweets

Pablo Gamallo, Marcos Garcia, Susana Sotelo
Centro de Investigación en Tecnoloxías da Información (CITIUS)
{*pablo.gamallo, marcos.garcia.gonzalez*}*@usc.es*

José Ramom Pichel
Imaxin|Software
jramompichel@imaxin.com

Resumen: En este artículo se describen dos sistemas que participaron en la competición TweetLID-2014, centrada en la detección del idioma en tweets. Los sistemas han sido desarrollados en base a dos estrategias: el uso de diccionarios ordenados por frecuencia y la construcción de clasificadores bayesianos. Los resultados muestran que los diccionarios con ranking funcionan mejor con pequeños corpus de entrenamiento cuya distribución por lengua es similar a la del conjunto de test, mientras que el algoritmo bayesiano mejora su eficacia con el uso de grandes modelos de entrenamiento incluso cuando la distribución de los datos que contienen es diferente al corpus de test. Los experimentos realizados también muestran que los modelos con unigramas de palabras funcionan mejor que el uso de n-gramas de caracteres. En la evaluación final, el clasificador bayesiano obtuvo la primera posición entre todos los sistemas "unconstrained" (entrenados adicionalmente con fuentes externas) que participaron en la competición.
Palabras clave: Identificación de idioma, Textos cortos, Clasificadores bayesianos, Modelos basados en diccionarios

Abstract: This article describes two systems participating to the TweetLID-2014 competition focused on language detection in tweets. The systems are based on two different strategies: ranked dictionaries and Naive Bayes classifiers. The results show that ranking dictionaries performs better with small training corpora whose language distribution is similar to that of the test dataset, while a Naive Bayes algorithm improves the scores with large models even if the data are unbalanced with regard to the test dataset. The experiments also showed that the models based on word unigrams outperform the use of n-grams of characters. In the final evaluation the Naive Bayes classifier got the first position among the unconstrained systems (trained with external sources) participating in the competition.
Keywords: Language Identification, Short Text, Naive Bayes Classifier, Dictionary-Based Models

1 Introduction

McNamee (2005) argued that language detection is a solved problem since the performance of most systems approaches 100% accuracy. However, this can be true only if we assume that the systems are tested on relatively long and well written texts. In recent experiments, the accuracy of the language detection starts to decrease much faster with respect to relatively longer texts having at least 400 characters.(Tromp and Pechenizkiy, 2011). In consequence, language detection is not a solved problem if we consider noisy short texts such as those written in social networks. Apart from the size and the written quality of input texts, it is also necessary to take into account another important factor that can hurt the performance of language detectors, namely language proximity. Closely related languages are more difficult

* Work funded by HPCPLN - Ref:EM13/041 (Xunta de Galicia), Celtic - Ref:2012-CE138 y Plastic - Ref:2013-CE298 (Programa Feder-Innterconecta)

to identify and separate than languages belonging to different linguistic families.

TweetLID Competition (Zubiaga et al., 2014) is aimed to compare language detection systems tested on tweets written in the 5 most spoken languages from the Iberian Peninsula (Basque, Catalan, Galician, Spanish, and Portuguese), and English. Some of the target languages are closely related: e.g. Spanish and Galician or Spanish and Catalan, and even there are varieties of the same language in two different spelling rules, e.g. Portuguese and Galician. So the systems are tested, not only on noisy short texts (tweets), but also on a set of texts written in very similar languages/varieties. In addition, the systems must also identify those cases where the language cannot be determined: other language, interjections, etc. It is worth noting that this competition does not provide any supervised information on tweets, such as the language profile of the author. This type of information cannot be used by the participants, even if it is used by recent approaches to language identification in microblog posts (Carter, Weerkamp, and Tsagkias, 2013).

In related work, two types of models have been used for language detection: those made of n-grams of characters (Beesley, 1988; Dunning, 1994) and those based on word unigrams or dictionaries (Grefenstette, 1995; Rehurek and Kolkus, 2009). In the latter approaches, models are dictionaries built with words ranked by their frequency in a reference corpus, and their ranking is used to compute their "relevance" in the input text. In Cavnar and Trenkle (1994), they construct a language model by making use of the ranking of the most frequent character n-grams for each language during the training phase (n-gram profiles). So, even if this is an approach based on character n-grams, it also uses the ranking strategy which is characteristic of the dictionary-based approach.

The objective of the article is to compare two methods for language detection in tweets. On the one hand, we describe a ranking approach based on small dictionaries built according to the Zipf's law, i.e. the frequency of any word is inversely proportional to its rank in the frequency table and, on the other hand, we also describe a Naive Bayes system which uses either n-grams of characters or word unigrams.

2 Two approaches

2.1 Quelingua: A Dictionary-Based Approach

Our system, called *Quelingua*[1], was implemented using a dictionary-based method and a ranking algorithm. It is based on the observation that for each language, there is a set of words that make up a large portion of any text and their presence is to be expected as word distribution follows Zipf's law.

For each word w found in a corpus of a particular language, and for the N most frequent words in that corpus, we define its *inverse ranking* (IR) as follows:

$$IR(w) = N - (rank(w) - 1) \qquad (1)$$

where $rank(w)$ is the rank of w in the dictionary of N most frequent words. For instance, if the dictionary contains 1000 words, the IR for the most frequent word (ranking 1) is 1000. Specifying the size N of the dictionary is a critical issue of the method. The final weight of a specific language *lang* given a text is computed in equation 2, where K is the size of the input text:

$$weight(lang, text) = \sum_{i=1}^{K} IR(word_i) \qquad (2)$$

This is computed for all available languages, and that with the highest weight is selected as the detected language for the input text.

In order to give more coverage to the system, we added a suffix module containing the most frequent suffixes of the target languages. For instance, "-ção" is associated to Portuguese, "-ak" to Basque, "-ción" to Spanish and Galician, etc. This information can be automatically extracted or manually added to the module. The IR of any word that is not in the dictionary but has a suffix found in the suffix module is computed as the average IR, i.e.: $N/2$.

2.2 A Naive Bayes Classifier

To compare our dictionary-based system with a state-of-the-art approach, we implemented a Naive Bayes (NB) classifier

[1]Freely available at: http://gramatica.usc.es/ gamallo/tools/quelingua.htm

based on the system we previously created for a sentiment analysis task, and described in Gamallo, Garcia, and Fernández-Lanza (2013). According to recent research (Winkelmolen and Mascardi, 2011; Vatanen, Väyrynen, and Virpioja, 2010), language detection based on NB algorithms performs well on very short texts. In Vatanen, Väyrynen, and Virpioja (2010), a NB classifier built with character n-gram models clearly outperformed the ranking method by Cavnar and Trenkle (1994) when the tests were performed on noisy short texts.

Our NB classifier was trained with two different models: a model based on character n-grams and another one based on word unigrams. The best character n-gram model turned out to be constituted by trigrams with also bigrams just for prefix and suffix positions. This is in accordance with previous research on NB classifiers for language detection where the best models were constituted by small n-grams, with $n < 4$ (Winkelmolen and Mascardi, 2011; Vatanen, Väyrynen, and Virpioja, 2010). The smoothing technique used by our classifier for unseen features (n-grams or words) is a version of Good-Turing estimation (Gale, 1995). As in Quelingua, frequent suffixes were also added as features to the model.

3 Experiments

3.1 Training and Test Dataset

To evaluate the performance of the two systems described above, the development corpus of tweets provided by the organization of TweetLID2014 was divided into two parts: 65% used for training and 35% as test dataset. In addition, the systems were also trained with further texts constituted by recent news extracted from online journals for English (11Mb), Spanish (7.3Mb), Portuguese (6.6Mb), and Galician (4.2Mb). The Catalan texts were taken from the Ancora corpus (Taulé, Martí, and Recasens, 2008) (2.2Mb) and the Basque corpus was compiled from 5 fictional and technical books (1.05Mb). For this preliminary experiments, the *constrained* systems were trained with the 65% of tweets of the development dataset, while the *unconstrained* systems were trained with those tweets as well as the external text corpora.

3.2 Preprocessing

Before building the features used by the systems, the main preprocessing tasks we considered are the following: removing urls, references to usernames, hashtags, and emoticons; reduction of replicated characters for voyels (e.g. *loooveeee* $\rightarrow$ *love*) ; normalizing the text by using a small list of abbreviations (e.g. $x \rightarrow por$).

3.3 First Evaluations

To evaluate the two systems, we used in our experiments the evaluation script provided by the TweetLID-2014 organization. As far as the NB classifier is concerned, we performed some experiments with both the constrained and unconstrained training data, as well as with both character n-grams and word unigrams (bag of words). The best results were achieved with unconstrained training data and word unigrams. The highest F1-Score reached with character n-grams was $63,56\%$ using unconstrained training and $n < 4$. By contrast, the best results achieved with word unigrams was $77,94\%$ also using the unconstrained training. This is in accordance with Rehurek and Kolkus (2009), who tried to prove that dictionary-based methods are more reliable than character-based systems for language identification with noisy short texts among similar languages. In the following experiments, we will only use word unigrams with the NB approach.

Concerning the dictionary-based system (Quelingua), the results obtained with the constrained training data clearly outperformed those obtained with the unconstrained version. Then, for the constrained system, we performed some experiments focused on determining the best size of the dictionary (i.e. of the language model). Figure 1 depicts the growth curve of F1-Score as a function of the size of the dictionary. It shows that the peak is achieved with a size of 1000 words. In the following experiments, Quelingua was trained with a dictionary of this size.

3.4 Results

Table 1 shows the results obtained by our two classifiers, *NB* and *Que(lingua)*, using different resources to train the model: only training tweets (constrained), only external resources (external), and both tweets and external resources (unconstrained).

The best constrained system is Quelingua

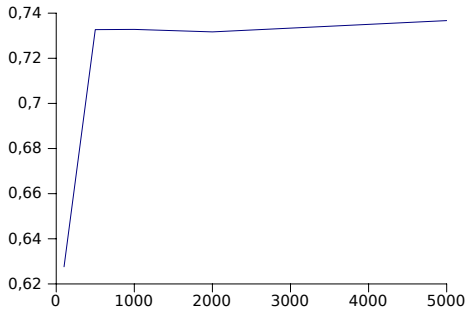

Figure 1: Growth curve of F1-Score (y axis) as a function of the dictionary size (x axis)

while the best unconstrained is NB (which also reaches the highest score overall). Unlike Quelingua, the NB system achieves the best results with the unconstrained model. The behavior of NB is not only different from Quelingua, but also from the other systems participating in TweetLID-2014 competition, since all systems except NB perform better with the constrained version. We cannot afford a full explanation for the other systems in the competition, but the main reason of the NB behavior with regard to Quelingua is that NB-based systems tend to perform better with large model sizes than classifiers based on ranking methods (Vatanen, Väyrynen, and Virpioja, 2010). This way, as it has been observed above, Quelingua (which is a ranking method) requires small vocabularies that can be learned from small text corpora. Another key factor is language distribution. Ranking methods work well with few training data but they are quite sensitive to the language distribution. Their performance decreases significantly when the language distribution of the test dataset is very different from that of the training set (as in the unconstrained model). By contrast, NB models can mitigate unbalanced distribution with more training data. In sum, Quelingua works better with small but balanced training corpora while NB reaches higher scores with large (even if unbalanced) corpora.

However the two systems behave in a similar way when they are observed across the different target languages. Both systems reach acceptable results (between 85 and 95% F-Score) in Portuguese, English, Spanish, and Catalan, and poor results in Basque, Galician, and Und(efined).

Four runs were sent to the final TweetLID-2014 evaluation: the constrained and unconstrained versions of both NB and Quelingua trained with the whole training dataset. The unconstrained version of NB achieved the highest score among all participants (75.3% F1-score). The constrained version of Quelingua achieved the fourth position out of 12 runs (72.6% F1-score) . It is worth noting that the final results obtained with the test dataset follow a similar tendency as that observed in our previous experiments (Table 1).

3.5 Efficiency

In terms of memory use, Quelingua loads a light dictionary of 35Kb (1000 words per language), while the NB systems requires loading a language model of 9Mb. Concerning speed, classification based on NB models is much slower than classification with the ranking method of Quelingua. More precisely, Quelingua is about 10 times faster than NB.

4 Conclusions and Future Work

We compared several strategies for language detection in noisy short messages (tweets). First, we observed that models with word unigrams perform better than those based on n-grams of characters. We also observed that our Naive Bayes classifier outperforms the ranking-based method (Quelingua) if they are trained with external corpus (unconstrained models). By contrast, the ranking method performs better than NB when they use a small training set of tweets containing similar data (and same language distribution) to the test dataset (constrained model). Besides the fact of performing reasonably well with a small and balanced training corpus, another benefit of the ranking model is its small and easy to handle ranked dictionary, which can be easily corrected and updated by human experts.

In fact, in future work, we will measure the performance effects of using a manually corrected ranked vocabulary, since the dictionaries used in the described experiments were not corrected by humans. We will also analyze the growth curve of the F1-score obtained by the NB system over the corpus size. Finally, it will be interesting to compare these approaches with contextual-based strategies such as Markov Models, which were the best systems according to other evaluations (Padrò and Padrò, 2004).

Lang	NB-cons	NB-extern	NB-uncons	Que-cons	Que-extern	Que-uncons
es	92.95	92.74	**94.54**	91.18	85.28	88.62
pt	92.42	85.71	**93.45**	92.23	71.50	82.01
en	83.56	82.05	84.22	**85.92**	81.41	80.76
ca	**92.95**	89.41	92.68	87.57	76.87	80.63
eu	50.54	65.36	**71.02**	63.39	57.71	63.15
gl	48.11	64.61	**66.20**	53.17	47.47	50.29
und	27.82	28.57	28.84	**29.71**	19.44	21.79
average	72.66	74.48	**77.94**	73.28	63.86	67.29

Table 1: F-Score reached by both NB and QUE(lingua) systems when trained with three different resources

References

Beesley, Kenneth R. 1988. Language identier: A computer program for automatic natural-language identication of on-line text. In *29th Annual Conference of the American Translators Association*, pages 47–54.

Carter, S., W. Weerkamp, and M. Tsagkias. 2013. Microblog language identification: Overcoming the limitations of short, unedited and idiomatic text. *Language Resources and Evaluation*, 47:195–215.

Cavnar, William B. and John M. Trenkle. 1994. N-gram-based text categorization. In *Proceedings of the Third Symposium on Document Analysis and Information Retrieval*, Las Vegas, USA.

Dunning, Ted. 1994. *Statistical Identification of Language*. Technical Report MCCS 94–273. New Mexico State University.

Gale, Willian. 1995. Good-turing smoothing without tears. *Journal of Quantitative Linguistics*, 2:217–37.

Gamallo, Pablo, Marcos Garcia, and Santiago Fernández-Lanza. 2013. TASS: A Naive-Bayes strategy for sentiment analysis on Spanish tweets. In *Workshop on Sentiment Analysis (TASS2013)*, pages 126–132, Madrid, Spain.

Grefenstette, Gregory. 1995. Comparing two language identification schemes. In *International Conference on the Statistical Analysis of Textual Data (JADT 1995)*.

McNamee, Paul. 2005. Language identification: a solved problem suitable for undergraduate instruction. *Journal of Computing Sciences in Colleges*, 3:94–101.

Padrò, Muntsa and Lluís Padrò. 2004. Comparing methods for language identification. *Procesamiento del Lenguage Natural*, 33:151–161.

Rehurek, Radim and Milan Kolkus. 2009. Language identification on the web: Extending the dictionary method. *Lecture Notes in Computer Science*, pages 315–345.

Taulé, M., M.A. Martí, and M. Recasens. 2008. Ancora: Multilevel annotated corpora for catalan and spanish. In *LREC-2008.*, Marrakesh, Morocco.

Tromp, Erik and Mykola Pechenizkiy. 2011. Graph-based n-gram language identification on short texts. In *Proceedings of Benelearn 2011*, pages 27–35, The Hague, Netherlands.

Vatanen, Tommi, Jaakko J. Väyrynen, and Sami Virpioja. 2010. Slanguage identification of short text segments with n-gram models. In *Proceedings of LREC-2010*.

Winkelmolen, Fela and Viviana Mascardi. 2011. Statistical language identification of short texts. In *Proceedings of ICAAR*, pages 498–503.

Zubiaga, Arkaitz, Iñaki San Vicente, Pablo Gamallo, José Ramom Pichel, Iñaki Alegria, Nora Aranberri, Aitzol Ezeiza, and Víctor Fresno. 2014. Overview of tweetlid: Tweet language identification at sepln 2014. In *TweetLID @ SEPLN 2014*, Girona, Spain.