Tuning Bayes Baseline for dialect detection

Hector-Hugo Franco-Penya
Dublin Institute of Technology
francoph@tcd.ie

Liliana Mamani Sanchez
University College Dublin
mamanisl@tcd.ie

Abstract

This paper describes an analysis of our submissions to the Dialect Detection Shared Task 2016. We proposed three different systems that involved simplistic features, to name: a Naive-bayes system, a Support Vector Machines-based system and a Tree Kernel-based system. These systems underperform when compared to other submissions in this shared task, since the best one achieved an accuracy of $\sim$0.834.

1 Introduction

The problem of discriminating similar languages has been tackled in previous years in the context of shared tasks (Zampieri et al., 2014; Zampieri et al., 2015b). Here, there are promising results for dialect detection, being the best results around 95.54% and for an open challenge, the best results yield around 95.65%.

Despite these positive results, some research issues remain to be solved such as: domain adaptation, inclusion of new languages and classifier performance in terms of processing time.

The DSL 2014 Shared Task aimed to discriminate dialects within each of these 6 groups: Group A (Bosnian, Croatian, Serbian), Group B (Indonesian, Malay), Group C (Czech, Slovak), Group D (Brazilian Portuguese, European Portuguese), Group E (Castilian Spanish, Argentine Spanish), and Group F (American English, British English). The 2015 version of this Shared task considered the first 5 groups plus an additional group comprising Bulgarian and Macedonian. The 2016 version (Malmasi et al., 2016) where our systems were competing differs to previous tasks in the addition of a new variety of Spanish language: Mexican Spanish. Additionally, a second task aims to test dialect identification systems in Arabic language datasets.

Our submissions mainly addressed the sub-task 1 for automatically discriminating between similar languages and language varieties. We took into account two principles: a) to design lightweight systems given that such a system should work in an online environment, and b) to design systems that would involve using grammatical information without the recurring to sophisticated parsers.

This paper is structured as follows: Section 2 provides a brief context for our work in the state of the art of language detection. Section 3 describes the core of our experiments. Section 4 outlines our results and an analysis of the relevance of the proposed methods. Finally, we conclude with Section 6.

2 Related Work

Zampieri et al. (2012) and Zampieri et al. (2013) developed a computationally efficient method for detecting Spanish and French dialects, which produces highly accurate results based on a naive Bayes method. They address dialection detection in text extracted from newspaper articles. Since one of our systems is mainy based in this method we provide a more detailed explanation in Section 3.3.

The previous DSL shared task was hold in 2015, in for which nine systems where submitted to the close challenge. The best system was developed by (Malmasi and Dras, 2015). It consists of an ensemble of SVM classifiers trained with character ngrams and word unigrams and bigrams.

Other approaches were based on two stage classification, the first stage was designed to classify the group of languages and the second stage to differentiate between dialects (Goutte and Serge, 2015; Fabra-

Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects,
pages 227–234, Osaka, Japan, December 12 2016.

Boluda et al., 2015; Acs et al., 2015). Zampieri et al. (2015a) created a system based on Support Vector Machines that has features in form of TF-IDF and also token base back-off (Jauhiainen et al., 2015), an interesting method that split unknown tokens (unknown in the training data sets) into character n-grams until it is found some examples on the training data set that can be use to derived probabilities.

3 Methods

This section briefly describes the resources and methods used to develop our systems.

3.1 Datasets

The training datasets provided by the shared task organizers were created based on text from newspapers articles. One in-domain test set and also two out of domain twitter base data sets were made available for testing purposes; these two twitter data sets were collected in a different manner to the news dataset.

3.2 Pre-processing

Punctuation marks, brackets, parenthesis, hyphens, and multiple blank spaces were removed. Also, sentences were standardized to be all in upper-case. This pre-processing simplifies the text and that could be beneficial on classification tasks with scarce amount of training data, but could also lose relevant information for the classification, for instance (Tan et al., 2012) claim that in Malaysian numbers are written with decimal point while in Indonesian are written using colons.

3.3 Naive Bayes, bi-gram language model

Our best system is a re-creation of the lightweight naive bayes bi-gram-word classification model described in (Zampieri et al., 2015a; Zampieri et al., 2013; Zampieri et al., 2012; Zampieri and Gebre, 2012; Tiedemann and Ljubešić, 2012; Baldwin and Lui, 2010) for detecting Spanish dialects, Portuguese dialects (from Brazil or Portugal), between Bosnian, Croatian and Serbian, and other languages. This model has been extensively tested in different scenarios in the aforementioned works and we deemed it was a good starting point for our experiment and it seemed less demanding in terms of processing times. Its implementation was also described in language identification studies (Tiedemann and Ljubešić, 2012).

The formula used to calculate the likelihood of a given text belonging to a language or dialect L is:

$$P(L|text) = argmax_L \sum_{i=1}^{N} log(P_l(n_i|L)) + log(P(L))) \quad (1)$$

where N is the number of n-grams, $P_l(n_i|L)$ is the Laplace probability of the n_i n-gram appearing on the language model L and $P(L)$ is the ratio of the number of n-grams used to build the language model L divided by the total amount of n-grams used to build all language models.

$$P_l(ng|L) = \frac{C(ng|L) + \alpha}{N + B} \quad (2)$$

where $\alpha = 1$. $C(ng|L)$ is the number of times the n-gram ng appears on the text used to build the language model L. N is the total number of n-grams extracted from the text used to build L, and B is the total number of unique n-grams found at the text used to build the language model L.

The best results on the discerning western languages development data set where reach using bi-grams, therefore bi-grams models where used to for both tasks.

3.4 SVM

Support Vector Machines are among the most used algorithms for classification problems. Baldwin and Lui (2010) successfully used SVMs in language identification. It was also used in previous shared tasks in different setups (Purver, 2014; Zampieri et al., 2015a; Malmasi and Dras, 2015).

Each unique word on the train data set was assigned a unique index. Using these indexes, a sparse vector was created for each sentence of the training and testing data set. Words which did not appear on the training data set were ignored. The appearance of a word was flagged as a single occurrence

on the projected vector independently of how many times that word appeared on the sentence. For this experiment the multi-class setup of lib-SMV was used.

4 Analysis of Results

This shared task is about classifying sentences, and context plays a crucial role. Nonetheless, the authors think it is worth to discuss the importance of dialect detection when the dialect of a short piece of language cannot be detected by neither humans or machines.

Two scenarios appear likely: a) such piece of language is standard amongst language variations and understood by the great majority of native speakers of the corresponding language, or b) It is too specific in a dialect and in a register within that dialect that there is there is no body of comparison that allows detection.

For the second case, let us consider the domain register of the dataset used during the training phase of the experiments.

4.1 Training times

The naive Bayes model is quickly trained because it just requires to calculate n-grams probabilities and has a linear computational cost (see Table 1). SVM is has a quadratic computational time, and the training time is measured in hours or minutes, except for the Arabic data set, which is measured in seconds due to its small size.

All experiments were done in a laptop with an Intel Core i7-5600U processor at 2.60GHz with 2 Cores and 16GB of RAM.

Method	Sub-task	Set	unigrams	bigrams
SVM	1	train	5.5 hours	5 hours
SVM	2	train	30 seconds	12 seconds
SVM	1	dev	20 minutes	14 minutes
Bayes	1	train	11 seconds	17 seconds
Bayes	2	train	<1 second	<1 second
Bayes	1	dev	<1 second	<1 second

Table 1: Training times for both training datasets: Sub-task 1(for Roman Alphabet Languages) and Sub-task 2 (for the Arabic language)

5 Overview

The official results for our submissions are shown in Table 2. They correspond to two of the systems described in Section 3. This table shows the accuracy, micro and macro and weighted F1 for each of the submitted classifications. Test set A is the in-domain composed by text from newspaper articles. Test datasets B1 and B2 are composed by text extracted from twitter microposts, in two different ways. Test set C is composed by Arabic text extracted by Automatic Speech Recognition.

Test Set	**Run**	**Accuracy**	**F1 (micro)**	**F1 (macro)**	**F1 (weighted)**
A	run1(Bayes)	0.8377	0.8377	0.8317	0.8317
A	run2(SVM)	0.5848	0.5848	0.5802	0.5802
B1	run1(Bayes)	0.806	0.806	0.5667	0.7934
B1	run2(SVM)	0.594	0.594	0.3949	0.4739
B2	run1(Bayes)	0.74	0.74	0.4543	0.7268
B2	run2(SVM)	0.588	0.588	0.3394	0.543
C	run1(Bayes)	0.3584	0.3584	0.3492	0.3455

Table 2: Results for all runs (for the closed track)

Table 3 shows the confusion matrix for the in-domain test set A per language. When a word cannot be identified as belonging to any language model the system classifies as “bs” by default, that is why the

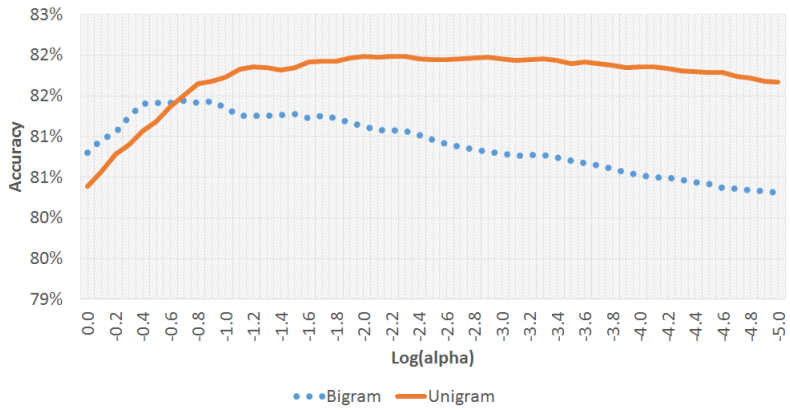

Figure 1: Accuracy graph for Lidstone smoothing factor (development data set)

first column (predicted bs) shows more false positives than other languages. This confusion table shows that the system classifies the language of a sentence with an accuracy of about 99.5%.

	bs+hr+sr	es	fr	id-my	pt
bs+hr+sr	2994	3	1	1	1
es	3	2994	1	1	1
fr	9	1	1989	0	1
id-my	7	2	2	1988	1
pt	3	1	1	0	1995

Table 3: Language confusion matrix

Table 4 shows the confusion matrix for all dialects. Table 5 shows Precision, Recall and F1 of each dialect, in domain test set. Here the Indonesian and Malasian (Group B) show the highest F1-scores, and Bosnian, Croatian and Serbian (group A) show the lowest F1-scores.

Table 6 shows the confusion matrix for the twitter testing data sets B1 and B2, the tables are simplified because there are only testing samples for group D:Portuguese and group A: Bosnian, Croatian and Serbian, rows corresponding to other dialects where removed as only contain zeros, but those dialects may appear in the column section to identify false positives, which in this case are es-es, fr-fr and id.

B1	bs	es-ar	es-es	es-mx	fr-ca	fr-fr	hr	id	my	pt-br	pt-pt	sr
bs	500		2				247					251
es-ar		861	126	12			1					
es-es	2	70	909	16		1		1			1	
es-mx		191	350	459								
fr-ca					863	137						
fr-fr	6		1		46	943	3				1	
hr	78		1				871					50
id	5	1	1			2		976	14		1	
my	2							40	958			
pt-br	1									945	54	
pt-pt	2		1			1				95	901	
sr	79					1	52	1		1		866

Table 4: Confusion matrix results

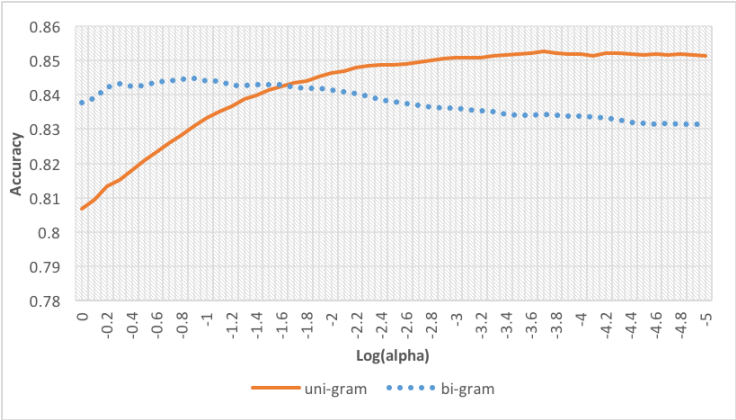

Figure 2: Accuracy graph for Lidstone smoothing factor for A (in-domain test).

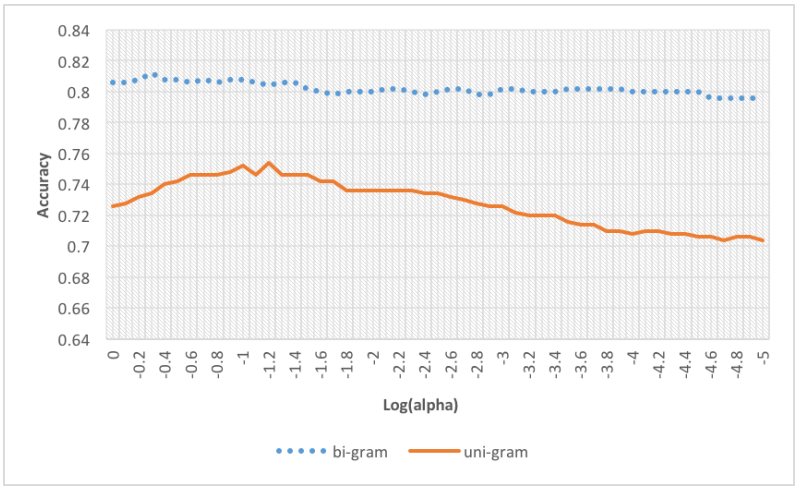

Figure 3: Accuracy graph for Lidstone smoothing factor for B1 (first out-of-domain twitter data set)

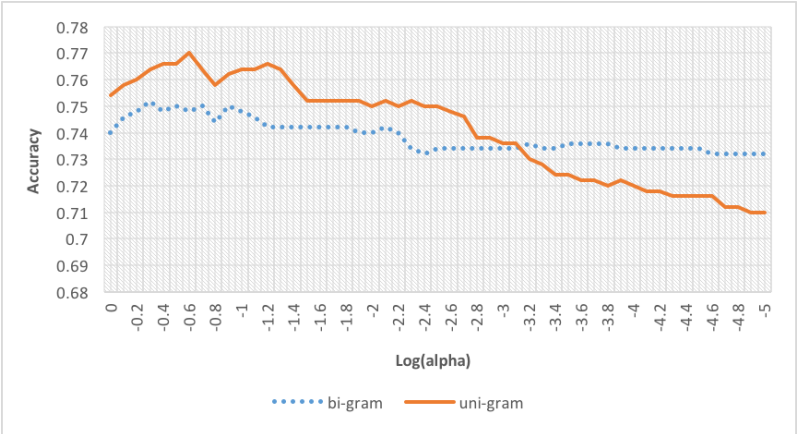

Figure 4: Accuracy graph for Lidstone smoothing factor for B2 (second out-of-domain twitter data set)

	bs	es-ar	es-es	es-mx	fr-ca	fr-fr	hr	id	my	pt-br	pt-pt	sr
Precision	50	86	91	46	86	94	87	98	96	95	90	87
Recall	43	43	40	49	49	46	43	49	50	48	48	43
F1	46	58	55	47	62	62	57	65	65	63	63	57

Table 5: Precision, Recall and F1 of each dialect, in domain test set (in percentages)

	Predicted							
	bs-hr-sr dialects			Portuguese		Others		
B1	bs	hr	sr	pt-br	pt-pt	es-es	fr-fr	id
bs	36	6	56	.	.	1	.	1
hr	2	90	8	.	.	.	.	.
sr	1	.	99	.	.	.	.	.
pt-br	.	.	.	99	1	.	.	.
pt-pt	.	.	.	21	79	.	.	.
B2	bs	hr	sr	pt-br	pt-pt	es-es	fr-fr	id
bs	34	7	57	.	.	1	.	1
hr	4	87	9	.	.	.	.	.
sr	1	.	99	.	.	.	.	.
pt-br	3	.	.	88	9	.	.	.
pt-pt	2	.	.	34	62	1	1	.

Table 6: Simplified confusion matrix for the twitter test datasets B1 and B2.

5.1 Lidstone smoothing factor

With Lidstone smoothing factor α set to one, the probability formula results in the Laplace probability, however with $\alpha < 1$ the probability results in Lidstone smoothing (Tan et al., 2012).

Extensive experimentation using Laplace probability has been carried out ($\alpha = 1$) (Baldwin and Lui, 2010; Tan et al., 2012; Zampieri and Gebre, 2012; Zampieri et al., 2013; Zaidan and Callison-Burch, 2014), but non as far as the authors of this article know using an optimization of Lidstone smoothing.

An investigation carried out after the submission deadline for the experiment using the development set, shows that an accuracy of 80.4% for uni-grams using Laplace probabilities could be improved to 82.0% (increment of 1.6%) with Lidstone smoothing using an alpha value of $\alpha = 0.01$.

The tuning of the parameter α seems to improve the uni-grams language model more than the bi-grams language model to the extent that the uni-gram model outperforms the bi-gram one. This is an important observation because it was believed that the bi-gram model outperforms the uni-gram model, and that is what happens with $\alpha = 1$, this is why the results of the bi-gram model where submitted to the shared task evaluation.

Figure 1 shows how the accuracy changes along with the parameter α. The smallest the α value the higher weight of infrequent words on the results of the experiment.

It is a plausible hypothesis that uni-gram models are less likely to capture name entities that consist on multiple words and therefore not only outperforms the bi-gram model but also adapts better to new domains. This expectation is not observed on the results show on other data sets, for instance Figure 2 shows a similar trend by the crossing point from which uni-grams outperform bi-grams $10^{-1.6}$ is much lower than the one suggested by the development set $10^{-0.6}$. Also the out of domains data sets, B1 shows a graph for which bi-grams always outperform uni-grams (See Figure 3), and B2, shows opposite trends as the in-domain testing set (see Figure 3).

6 Concluding Remarks

Observations on the tuning of the smoothing factor (Section 3.3) are important contributions of this work. This results indicates that with proper selection of the α parameter the word base uni-gram model tends

to outperforms the word base bi-gram model. This is important because previous published research used the default parameter $\alpha = 1$ and it looks like bi-gram word base models outperform uni-gram models, where the results shown on this article point otherwise. Uni-gram word base models can have smaller dictionaries which probably are less attached to the training set domain and that could lead to better domain adaptation, this hypothesis needs further investigation.

The optimal value for the alpha parameter seems to be substantially lower than the default set on Laplace probabilities, about $\alpha = 0.01$ for words uni-grams and $\alpha = 0.2$ for words bi-grams, where the crossing point from which the uni-grams model outperform the bi-grams model is $\alpha = 0.25$ $(10^{-0.6})$ this values are derivative from the development set.

This article re-produces a successfully naive Bayes language classifier approach for the automatic classification. The system was trained with for two different groups of languages, the first task contains twelve different languages or dialects group in five different clusters according to their similarity. The groups are: Group A (Bosnian, Croatian, and Serbian), Group B (Malay and Indonesian), Group C (Portuguese: Brazil and Portugal), Group D (Spanish: Argentina, Mexico, and Spain), Group E (French: France and Canada).

Classifying sentences among this groups of languages is not a novel task if analysed on individual groups but what is novel is to discriminate with all twelve groups together, except on previous shared tasks.

As an interesting observation using naive Bayes about 4.2% of the in-domain test set of Argentinian-Spanish is classified as Castilian-Spanish, where almost no Argentinian-Spanish samples are classified as Mexican-Spanish (0.4%). However with the SVM model, this trend is reversed, with SVM still 3% of Argentinian-Spanish samples are misclassified as Spanish, and 13% are misclassified as Mexican-Spanish.

Regarding the second task, classifying Arabic languages/dialects, the results obtained using naive Bayes differ in great manner from the naive Bayes system described in (Zaidan and Callison-Burch, 2014) where the accuracy for each dialect ranges between 69.1% to 86.5%. The data sets are not the same, but the difference could be due to a problem on encoding Arabic characters.

References

Judit Acs, László Grad-Gyenge, Thiago Bruno, Rodrigues de Rezende Oliveira, and Vale do Sao Francisco. 2015. A two-level classifier for discriminating similar languages. In *Proceedings of the Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects, LT4VarDial*, volume 15, pages 73–77.

Timothy Baldwin and Marco Lui. 2010. Language Identification: The Long and the Short of the Matter. *Proceedings of the 2010 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT '12)*, (June):229–237.

Raül Fabra-Boluda, Francisco Rangel, and Paolo Rosso. 2015. NLEL UPV Autoritas participation at Discrimination between Similar Languages (DSL) 2015 Shared Task. In *Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects*, page 52.

Cyril Goutte and Leger Serge. 2015. Experiments in Discriminating Similar Languages. In *Proceedings of the Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects 2015*, pages 78–84, Bulgaria.

Tommi Jauhiainen, Heidi Jauhiainen, Krister Lindén, and Others. 2015. Discriminating similar languages with token-based backoff. In *Proceedings of the Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects*.

Shervin Malmasi and Mark Dras. 2015. Language Identification using Classifier Ensembles. In *Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects*, pages 35–43.

Shervin Malmasi, Marcos Zampieri, Nikola Ljubešić, Preslav Nakov, Ahmed Ali, and Jörg Tiedemann. 2016. Discriminating between similar languages and arabic dialect identification: A report on the third dsl shared task. In *Proceedings of the 3rd Workshop on Language Technology for Closely Related Languages, Varieties and Dialects (VarDial)*, Osaka, Japan.

Matthew Purver. 2014. A Simple Baseline for Discriminating Similar Languages. *Proceedings of the First Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects*, pages 155–160.

Liling Tan, Marcos Zampieri, and Nikola Ljubeši. 2012. Merging Comparable Data Sources for the Discrimination of Similar Languages : The DSL Corpus Collection. In *Proceedings of the 7th Workshop on Building and Using Comparable Corpora: Building Resources for Machine Translation Research*, Reykjavik, Iceland.

Jörg Tiedemann and Nikola Ljubešić. 2012. Efficient Discrimination Between Closely Related Languages. *Coling 2012*, (December 2012):2619–2634.

Omar F. Zaidan and Chris Callison-Burch. 2014. Arabic dialect identification. *Computational Linguistics*, 40(1):171–202, March.

Marcos Zampieri and Binyam Gebrekidan Gebre. 2012. Automatic Identification of Language Varieties: The Case of Portuguese. In *Proceedings of KONVENS 2012*, pages 233–237.

Marcos Zampieri, Binyam Gebrekidan Gebre, and Sascha Diwersy. 2012. Classifying Pluricentric Languages: Extending the Monolingual Model on. *Proceedings of the Fourth Swedish Language Technlogy Conference (SLTC2012)*, pages 79–80.

Marcos Zampieri, Binyam Gebrekidan Gebre, and Sascha Diwersy. 2013. N-gram Language Models and POS Distribution for the Identification of Spanish Varieties. In *Proceedings of TALN 2013 (Volume 2: Short Papers)*, pages 580–587, Les Sables d'Olonne, France, June. ATALA.

Marcos Zampieri, Liling Tan, Nikola Ljubešić, Jörg Tiedemann, and Nikola Ljube. 2014. A Report on the DSL Shared Task 2014. *Proceedings of the First Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects*, (2013):58–67.

Marcos Zampieri, Binyam Gebrekidan Gebre, Hernani Costa, and Josef Van Genabith. 2015a. Comparing Approaches to the Identification of Similar Languages. *Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects (LT4VarDial'15). 2nd Discriminating between Similar Languages Shared Task (DSL'15)*, page 7.

Marcos Zampieri, Liling Tan, Nikola Ljubešić, Jörg Tiedemann, and Preslav Nakov. 2015b. Overview of the DSL Shared Task 2015. *Proceedings of the Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects*, (2014):1–9.