ASIREM Participation at the Discriminating Similar Languages Shared Task 2016

Wafia Adouane[1], Nasredine Semmar[2], Richard Johansson[3]
Department of FLoV, University of Gothenburg, Sweden[1]
CEA Saclay – Nano-INNOV, Institut CARNOT CEA LIST, France[2]
Department of CSE, University of Gothenburg, Sweden[3]
wafia.gu@gmail.com, nasredine.semmar@cea.fr
richard.johansson@gu.se

Abstract

This paper presents the system built by ASIREM team for the Discriminating between Similar Languages (DSL) Shared task 2016. It describes the system which uses character-based and word-based n-grams separately. ASIREM participated in both sub-tasks (sub-task 1 and sub-task 2) and in both open and closed tracks. For the sub-task 1 which deals with Discriminating between similar languages and national language varieties, the system achieved an accuracy of 87.79% on the closed track, ending up ninth (the best results being 89.38%). In sub-task 2, which deals with Arabic dialect identification, the system achieved its best performance using character-based n-grams (49.67% accuracy), ranking fourth in the closed track (the best result being 51.16%), and an accuracy of 53.18%, ranking first in the open track.

1 Introduction

Automatic Language Identification (ALI) is the task of identifying the natural language of a given text or speech by a machine. It is a necessary task to build any language-dependent system. ALI is a well-established Natural Language Processing (NLP) task for many languages which are well represented on the Web. Nowadays, the challenge, however, is the identification of languages which are not well-represented on the Web, also called under-resourced languages, as well as the discrimination between similar languages (DSL) and language varieties (DLV).

The DSL Shared task 2016 consists of two sub-tasks; sub-task 1 and sub-task 2. Sub-task 1 (discriminating between similar languages and national language varieties) deals with twelve languages and language varieties grouped by similarity into 5 groups (Bosnian (bs), Croatian (hr), and Serbian (sr); Malay (my) and Indonesian (id); Portuguese: Brazil (pt-br) and Portugal (pt-pt); Spanish: Argentina (es-ar), Mexico (es-mx), and Spain (es-es); French: France (fr-fr) and Canada (fr-ca)). Sub-task 2 deals with Arabic dialect identification (Malmasi et al., 2016) including five Arabic varieties, namely Egyptian (EGY), Gulf (GLF), Levantine (LAV), North-African (NOR), and Modern Standard Arabic (MSA). We participated in both sub-tasks and we submitted four runs for both closed and open tracks (two for each). We trained a linear Support Vector Machines (SVM) classifier using both character-based and word-based n-grams as features.

The paper is organized as follows: in Section 2, we briefly describe some related work done for DSL and DLV tasks for both Arabic and other languages. In Section 3, we describe our system and the different runs we submitted. Then we present our results for each run in Section 4. We conclude by discussing the results and providing some suggestions to improve the current system.

2 Related Work

Discriminating between Similar Languages (DSL) and Discriminating between Language Varieties (DLV) are one of the serious bottlenecks, among others, of current automatic language identification tools. They are an even bigger challenge for under-resourced languages. DLV is a special case of DSL

Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects,
pages 163–169, Osaka, Japan, December 12 2016.

where the languages to distinguish are very close. These tasks have recently attracted the intention of the research community, resulting in recurring competitions such as the DSL Shared Task (Goutte et al., 2016). DSL can be simply defined as a specification or a sub-task of Automatic Language Identification (ALI) (Tiedemann and Ljubešić, 2012). Many of the standard methods used for the ALI have been applied to the DSL and DLV tasks for some languages. Goutte et al. (2016) give a comprehensive bibliography of the recently published papers dealing with these tasks. Discriminating between Arabic varieties is also an active research area although limited work has been done, so far, to distinguish between written Arabic varieties. The main reason is the lack of annotated data (Benajiba and Diab, 2010). Zaidan (2012) in his PhD distinguished between four Arabic varieties (Modern Standard Arabic (MSA), Egyptian, Gulf and Levantine dialects) using character and word n-gram models. Elfardy and Diab (2013) identified MSA from Egyptian at a sentence level, Tillmann et al. (2014) proposed an approach to improve classifying Egyptian and MSA at a sentence level, and Saâdane (2015) in her PhD distinguished between Maghrebi Arabic (Algerian, Moroccan and Tunisian dialects) using morpho-syntactic information. Furthermore, Malmasi et al. (2015) used a parallel corpus to distinguish between six Arabic varieties, namely MSA, Egyptian, Tunisian, Syrian, Jordanian and Palestinian.

Distinguishing between spoken Arabic varieties is also an active research area as there are sufficient phone and TV program recordings which are easy to transcribed. "The problem is somewhat mitigated in the speech domain, since dialectal data exists in the form of phone conversations and television program recordings, but, in general, dialectal Arabic data sets are hard to come by" (Zaidan and Callison-Burch, 2014). Akbacak et al. (2009), Akbacak et al. (2011), Lei and Hansen (2011), Boril et al. (2012), and Zhang et al. (2013) are some examples of work done to distinguish between spoken Arabic varieties.
Similarly to Goutte and Léger (2015), we experimented with both character-based and word-based n-grams as features. However, we used only one prediction step instead of two for both sub-tasks. Compared to the system proposed by Malmasi and Dras (2015), we used the same set of features with only one SVM classifier instead of an ensemble of SVM classifiers.

3 Methodology and Data

We used a supervised machine learning approach where we trained a linear SVM classifier (LinearSVC) as implemented in the Scikit-learn package[1]. In sub-task 1, we submitted two runs (run1 and run2). We experimented with different character-based and word-based n-grams and different combinations as features, and we reported only the best scoring features for each run. In run1: we used character-based 4-grams as features using TF-IDF weighting scheme. In run2: we used word-based unigrams. In both runs, we trained only on the released training dataset (Tan et al., 2014), and we used the development set for evaluating the system and selecting the best performing features. Word-based unigrams scored better than word based bigrams and trigrams and character-based 4-grams outperformed the rest of n-grams.

In sub-task 2 (Arabic dialects identification), we also submitted two runs for the closed track (run1 and run2) and two others for the open track (run3 and run4). In run1 and run3, we used a combination of character-based 5-grams and 6-grams as features using TF-IDF. In run2 and run4, we used word-based unigrams also weighted by TF-IDF. In all cases, we did not introduce any data pre-processing or Named Entity (NE) filtering. For sub-task 2, the released training data (Ali et al., 2016) consisted of ASR transcriptions of conversational speech in five Arabic varieties which are Egyptian (EGY), Gulf (GLF), Levantine (LAV), North-African (NOR), and Modern Standard Arabic (MSA). We noticed that the released training data contained many inconsistencies such as incomplete sentences, the use of different labels for the same sentence (chunk of text) and many speech segmentation errors. We did not have enough time to properly deal with these issues. All we did was clean the data by removing duplicate sentences having different labels. For training and evaluation, we trained our system on 80% of the released training dataset and we used the remaining data (20%) as a development set because the released data for this sub-task did not include a development set. Likewise, we evaluated the system and selected the best performing features, namely Word-based unigrams and the combination of character-based 5-grams and 6-grams.

[1]For more information see: http://scikit-learn.org/stable/.

In run3 and run4 (open track) we trained on a new dataset containing 18,000 documents (609,316 words in total) collected, manually by native speakers, from social media (real world data). The documents were originally written by users in Arabic script and we transliterated them using the Buckwalter Arabic transliteration scheme. This dataset contains 2,000 documents for each of the eight most popular high level[2] Arabic varieties (Algerian (ALG), Egyptian (EGY), Gulf (GLF), Levantine (LAV), Mesopotamian (KUI), Moroccan (MOR), Tunisian (TUN) dialects and MSA) plus Arabicized Berber[3]. The dataset was built as part of a Master's thesis project in Language Technology (Adouane, 2016), and is freely available for research from the first author.

4 Results

In sub-task 1, in both run1 and run2, we tested our system on test set A (closed track). Results are shown in Table 1.

Run	Baseline	Features	Accuracy	F1 (micro)	F1 (macro)	F1 (weighted)
run1	0.083	char 4-grams	**0.8779**	**0.8779**	**0.8778**	**0.8778**
run2	0.083	word unigrams	0.8717	0.8717	0.8714	0.8714

Table 1: Results for test set A (closed training).

The results show that Character-based 4-grams model (run1) scores slightly better than the word-based unigram model (run2), giving 0.8779 and 0.8717 accuracy respectively. Both models outperform the random baseline. Figure 1 shows the confusion matrix of the system as described in run1.

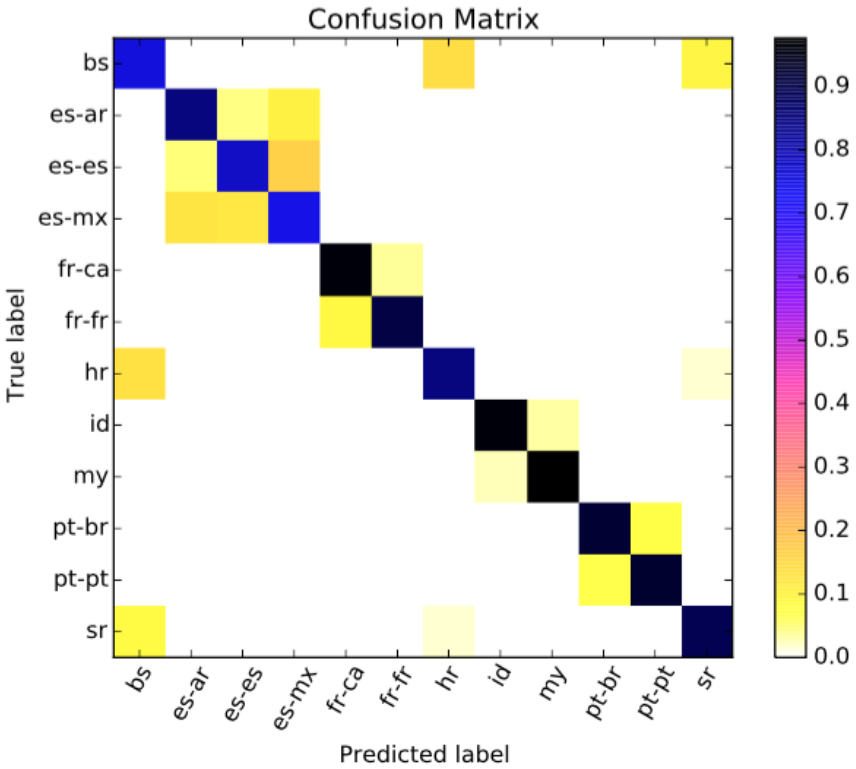

Figure 1: The confusion matrix of the system as in run1 (Table 1).

[2]We grouped local and regional varieties in one high level group.

[3]Berber or Tamazight is an Afro-Asiatic language widely spoken in North Africa and completely different from Arabic. It has 13 varieties and each has formal and informal forms. It has its unique script called Tifinagh but for convenience Latin and Arabic scripts are also used. Using Arabic script to transliterate Berber has existed since the beginning of the Islamic Era, see (Souag, 2004) for details.

The system is confused mostly between Spanish of Mexico and between Spanish of Argentina and Spanish of Spain. There is also confusion between Bosnian, Croatian and Serbian. Portuguese of Brazil is also confused with Portuguese of Portugal. Likewise, French of France is confused with French of Canada. Some confusions are also found between Indonesian and Malay. More or less, there is a confusion between all languages of the same group. The confusion is expected because those languages or language varieties are very similar.

As mentioned above, we participated in both closed and open track in sub-task 2 where we tested our system on the test set C and submitted two runs for each track. Table 2 and Table 3 show the evaluation results for the closed and open track respectively.

Run	Baseline	Features	Accuracy	F1 (micro)	F1 (macro)	F1 (weighted)
run1	0.2279	char 5+6-grams	**0.4968**	**0.4968**	**0.4914**	**0.4946**
run2	0.2279	word unigrams	0.4721	0.4721	0.4667	0.4711

Table 2: Results for test set C (closed training).

Run	Baseline	Features	Accuracy	F1 (micro)	F1 (macro)	F1 (weighted)
run3	0.2279	char 5+6-grams	**0.5318**	**0.5318**	**0.5255**	**0.5274**
run4	0.2279	word unigrams	0.4948	0.4948	0.4882	0.4912

Table 3: Results for test set C (open training).

The reported baseline in both tables is the majority class baseline because the samples in test set C were slightly unbalanced. It is clear that the combination of the character-based 5 and 6 grams scores better than the word-based unigram model in both closed and open tracks. The classification results outperformed the set baseline. The use of extra training dataset has improved the performance of the classifier compared to the use of the only provided training dataset.

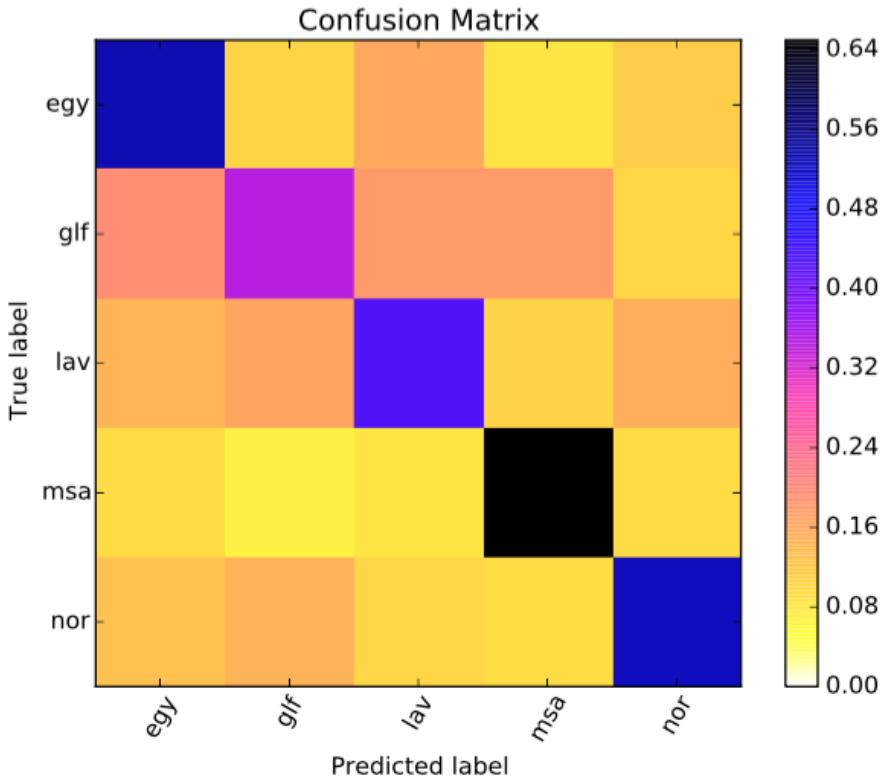

Figure 2: The confusion matrix of the system as in run1 (Table 2).

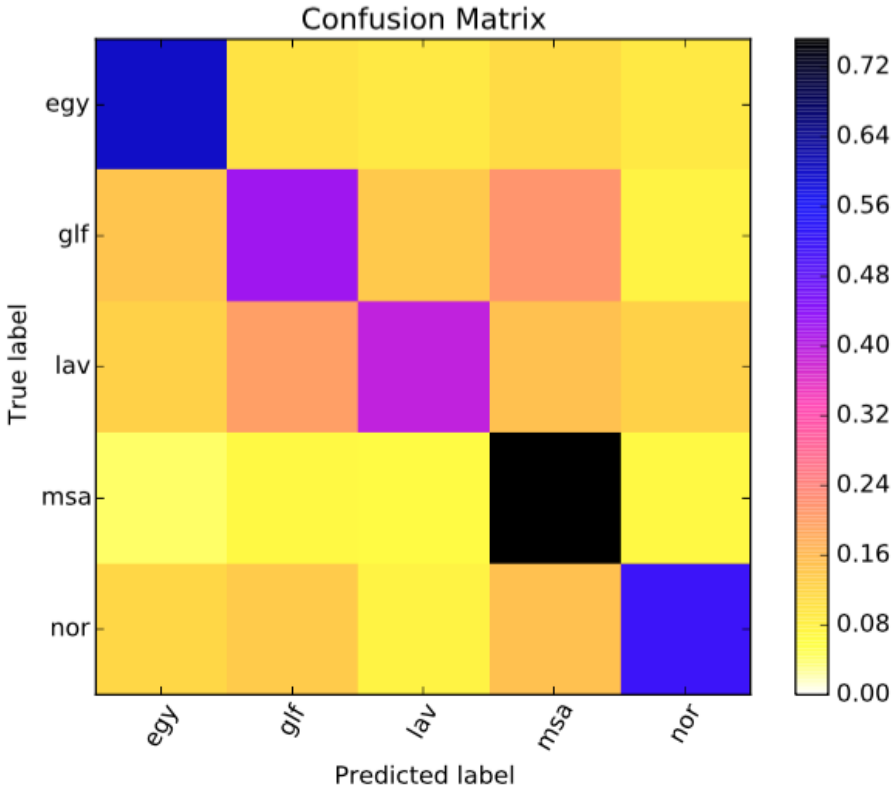

Figure 3: The confusion matrix of the system as in run3 (Table 3).

As shown in Figure 2 and Figure 3, the system misclassified all Arabic varieties with each other with different confusion degrees. Gulf Arabic is the most variety for which most mistakes are made, while MSA is the one that is most accurately recognized. Comparing between Figure 2 and Figure 3 shows that using extra training data has reduced the classification confusion in most cases, except for Levantine Arabic which is more confused with Gulf Arabic. This causes the number of correctly classified Levantine instances to decrease. It is also noticeable that there are more instances of all Arabic dialects confused with MSA. The results are expected as all these Arabic varieties use the same script with considerable vocabulary overlap and lots of false friends. Moreover, in the closed training, the used training dataset is very small.

5 Discussion

We have described our four submissions to the DSL Shared Task 2016 and presented the obtained results. We participated with the same system with no data preprocessing in both sub-task 1 and sub-task 2. Distinguishing between Arabic varieties (sub-task 2) is obviously more challenging than distinguishing between the languages included in sub-task 1. The main reason is of course related to the difference in linguistic properties between Arabic (all varieties included) and other languages. But most importantly, it is related to the quality of the data used in both training and evaluation. As mentioned above, the provided training data has many issues. Training the system on a larger manually collected dataset from social media domain (originally written texts in Arabic script) did not have a great effect on the performance of the system, especially that the test data (set C) consists of ASR transcripts which have many speech segmentation issues. It is worth mentioning also that we manually transliterated the new training dataset from Arabic script into Latin script (general replacement by mapping between letters using TextEdit) without any checking. There are some freely available scripts to do the transliteration automatically, but we preferred not to use them because of many encoding problems. The use of TF-IDF helped to get rid of most frequent (non-informative) words but only those seen in the training data which was very

small in our case. Still we believe that the proposed system is very simple. There are many possible improvements which can be done, for instance combining character and word-based n-grams, the use of dialectal lexicons as extra resources, the filtering of Named Entities (NE) because they are dialect or region specific. Another possible improvement is the removal of all MSA stop-words because MSA vocabulary is used in all other Arabic varieties. However, before that, we need to improve the quality of the training/evaluation data to allow the system to learn better language models.

References

Wafia Adouane. 2016. Automatic detection of under-resourced languages: The case of Arabic short texts. Master's thesis, University of Gothenburg.

Murat Akbacak, Horacio Franco, Michael Frandsen, Saša Hasan, Huda Jameel, Andreas Kathol, Shahram Khadivi, Xin Lei, Arindam Mandal, Saab Mansour, Kristin Precoda, Colleen Richey, Dimitra Vergyri, Wen Wang, Mei Yang, and Jing Zheng. 2009. Recent advances in SRI's IraqComm TM Iraqi Arabic-English speech-to-speech translation system. In *Proceedings of IEEE ICASSP*, pages 4809–4813, Taipei.

Murat Akbacak, Dimitra Vergyri, Andreas Stolcke, Nicolas Scheffer, and Arindam Mandal. 2011. Effective Arabic dialect classification using diverse phonotactic models. In *INTERSPEECH'11*, pages 4809–4813, Florence, Italy.

Ahmed Ali, Najim Dehak, Patrick Cardinal, Sameer Khurana, Sree Harsha Yella, James Glass, Peter Bell, and Steve Renals. 2016. Automatic dialect detection in arabic broadcast speech. In *Interspeech 2016*, pages 2934–2938.

Yassine Benajiba and Mona Diab. 2010. A web application for dialectal Arabic text annotation. In *Proceedings of the LREC Workshop for Language Resources (LRs) and Human Language Technologies (HLT) for Semitic Languages: Status, Up-dates, and Prospects*.

Hynek Boril, Abhijeet Sangwan, and John H. L. Hansen. 2012. Arabic dialect identification – Is the secret in the silence? and other observations. In *INTERSPEECH 2012*, Portland, Oregon.

Heba Elfardy and Mona Diab. 2013. Sentence Level Dialect Identification in Arabic. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL-13)*, Sofia, Bulgaria.

Cyril Goutte and Serge Léger. 2015. Experiments in Discriminating Similar Languages. In *Proceedings of the Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects (LT4VarDial)*, pages 78–84, Hissar, Bulgaria.

Cyril Goutte, Serge Léger, Shervin Malmasi, and Marcos Zampieri. 2016. Discriminating similar languages: Evaluations and explorations. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC)*.

Yun Lei and John H. L. Hansen. 2011. Dialect classification via text-independent training and testing for Arabic, Spanish, and Chinese. In *IEEE Transactions on Audio, Speech, and Language Processing, 19(1)*, pages 85–96.

Shervin Malmasi and Mark Dras. 2015. Language Identification using Classifier Ensembles. In *Proceedings of the Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects (LT4VarDial)*, pages 35–43, Hissar, Bulgaria.

Shervin Malmasi, Eshrag Refaee, and Mark Dras. 2015. Arabic Dialect Identification using a Parallel Multidialectal Corpus. In *Proceedings of the 14th Conference of the Pacific Association for Computational Linguistics (PACLING 2015)*, pages 209–217, Bali, Indonesia, May.

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.

Houda Saâdane. 2015. Le traitement automatique de l'arabe dialectalisé: aspects méthodologiques et algorithmiques. In *PhD thesis*, Université Grenoble Alpes, France.

Lameen Souag. 2004. Writing Berber Languages: a quick summary. In *L. Souag. Archived from http://goo.gl/ooA4uZ, Retrieved on April 8th, 2016*.

Liling Tan, Marcos Zampieri, Nikola Ljubešic, and Jörg Tiedemann. 2014. 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 (BUCC)*, pages 11–15, Reykjavik, Iceland.

Jörg Tiedemann and Nikola Ljubešić. 2012. Efficient Discrimination Between Closely Related Languages. In *Proceedings of COLING*, pages 2619–2634.

Christoph Tillmann, Yaser Al-Onaizan, and Saab Mansour. 2014. Improved Sentence-Level Arabic Dialect Classification. In *Proceedings of the 1st Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects*, pages 110–119, Dublin, Ireland.

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

Omar F. Zaidan. 2012. *Crowdsourcing Annotation for Machine Learning in Natural Language Processing Tasks*. Ph.D. thesis, Johns Hopkins University.

Qinghai Zhang, Hynek. Boril, and John. H. L. Hansen. 2013. Supervector Pre-Processing for PRSVM-based Chinese and Arabic Dialect Identification. In *IEEE ICASSP'13*, pages 7363–7367, Vancouver, Canada.