The George Washington University System for the Code-Switching Workshop Shared Task 2016

Mohamed Al-Badrashiny and Mona Diab
Department of Computer Science, The George Washington University
{badrashiny,mtdiab}@gwu.edu

Abstract

We describe our work in the EMNLP 2016 second code-switching shared task; a generic language independent framework for linguistic code switch point detection (LCSPD). The system uses characters level 5-grams and word level unigram language models to train a conditional random fields (CRF) model for classifying input words into various languages. We participated in the Modern Standard Arabic (MSA)-dialectal Arabic (DA) and Spanish-English tracks, obtaining a weighted average F-scores of 0.83 and 0.91 on MSA-DA and EN-SP respectively.

1 Introduction

Linguistic Code Switching (LCS) is a common practice among multilingual speakers in which they switch between their common languages in written and spoken communication. In Spanish-English for example: "She told me that mi esposo looks like un buen hombre." ("She told me that my husband looks like a good man"). In this work we care about detecting LCS points as they occur intra-sententially where words from more than one language is mixed in the same utterance. LCS is observed on all levels of linguistic representation. It is pervasive especially in social media. LCS poses a significant challenge to NLP, hence detecting LCS points is a very important task for many downstream applications.

In this shared task(Molina et al., 2016), the participants are asked to identify the language type of each word in a large set of tweets. The shared task has two language pair tracks; MSA-DA and Spanish-English. For each language pair, the participants are required to identify each word in each tweet to be:

- lang1: if the word is related to the first language in each track (i.e. MSA or English) ;
- lang2: if the word is related to the second language in each track (i.e. DA or Spanish);
- ambiguous: if the word can be in both languages and can't decide which language should be picked based on the context;
- mixed: if the word is consisted of mixed morphemes from both languages (ex. prefix and suffix form MSA attached to a DA word);
- fw: if the word is related to any other language than the targeted language pair
- ne: if the word is named entity;
- other: if the word is number, punctuation, emoticons, url, date, starts with #, @, or contains underscore;
- unk: if can not be determined to by any of the above tags.

Relevant work on thhe LCS problem among different language pairs can be summarized in the following work.

3ARRIB (Al-Badrashiny et al., 2014; Eskander et al., 2014) addresses the challenge of how to distinguish between Arabic words written using Roman script (Arabizi) and actual English words in the same context/utterance. The assumption in this framework is the script is Latin for all words. It trains

Proceedings of the Second Workshop on Computational Approaches to Code Switching, pages 108–111,
Austin, TX, November 1, 2016.

a finite state transducer (FST) to learn the mapping between the Roman form of the Arabizi words and their Arabic form. It uses the resulting FST to find all possible Arabic candidates for each word in the input text. These candidates are filtered using MADAMIRA (Pasha et al., 2014), a state of the art morphological analyzer and POS disambiguation tool, to filter out non-Arabic solutions. Finally, it leverages a decision tree that is trained on language model probabilities of both the Arabic and Romanized forms to render the final decision for each word in context as either being Arabic or English.

Bar and Dershowitz (2014) addresses the challenge for Spanish-English LCS. The authors use several features to train a sequential Support Vector Machines (SVM) classifier. The used features include previous and following two words, substrings of 1-3 character ngrams from the beginning and end of each word thereby modeling prefix and suffix information, a boolean feature indicating whether the first letter is capitalized or not, and 3-gram character and word n-gram language models trained over large corpora of English and Spanish, respectively.

Barman et al. (2014) present systems for both Nepali-English and Spanish-English LCS. The script for both language pairs is Latin based, i.e. Nepali-English is written in Latin script, and Spanish-English is written in Latin script. The authors carry out several experiments using different approaches including dictionary-based methods, linear kernel SVMs, and a k-nearest neighbor approach. The best setup they found is the SVM-based one that uses character n-gram, binary features indicates whether the word is in a language specific dictionary of the most frequent 5000 words they have constructed, length of the word, previous and next words, 3 boolean features for capitalization to check if the first letter is capitalized, if any letter is capitalized, or if all the letters are capitalized.

On the other hand, for within language varieties, AIDA2(Al-Badrashiny et al., 2015) is the best published system attacking this problem in Arabic for the Arabic varieties mix problem. In this context, the problem of LCS is more complicated than mixing two very different languages since in the case of varieties of the same language, the two varieties typically share a common space of cognates and often faux amis, where there are homographs but the words have very different semantic meanings, hence adding another layer of complexity to the problem. In this set up the assumed script is Arabic script. AIDA2 uses a complex system that is based on a mix of language dependent and machine learning components to detect the linguistic code switch between the modern standard Arabic (MSA) and Egyptian dialect (EGY) that are both written using Arabic script. It uses MADAMIRA(Pasha et al., 2014) to find the POS tag, prefix, lemma, suffix, for each word in the input text. Then it models these features together with other features including word level language model probabilities in a series of classifiers where it combines them in a classifier ensemble approach to find the best tag for each word.

In this paper we address this challenge using a generic simple language independent approach. We illustrate our approach on both language pair tracks.

2 Approach

The presented system in this paper is based on the idea we presented in (Al-Badrashiny and Diab, 2016). It is based on the assumption that each language has its own character pattern behaviors and combinations relating to the underlying phonology, phonetics, and morphology of each language independently. Accordingly, the manner of articulation constrains the possible phonemic/morphemic combinations in a language.

Accordingly, we use a supervised learning framework to address the challenge of LCS. We assume the presence of annotated code switched training data where each token is annotated as either Lang1 or Lang2. We create a sequence model using Conditional Random Fields (CRF++) tool(Sha and Pereira, 2003). For each word in the training data, we create a feature vector comprising character sequence level probabilities, unigram word level probabilities, and two binary features to identify if the word is named entity or not and is other or not . Once we derive the learning model, we apply to input text to identify the tokens in context. For the character sequence level probabilities, we built a 5-gram character language model (CLM) using the SRILM tool(Stolcke, 2002) for each of the two languages presented in the training data using the annotated words. For example, if the training data contains

	lang1	lang2	mixed	ne	ambiguous	fw	other	unk
MSA-DA-Training	127626	21722	16	21389	1186	0	13738	0
MSA-DA-Dev	6406	9326	2	3024	10	0	1888	0
EN-SP-Training	58844	27064	44	2364	252	11	20705	153
EN-SP-Dev	7067	5207	8	368	22	0	3912	58

Table 1: Language distribution (words/language) in the training and test data sets for all language-pairs

the two languages "lang1" and "lang2", we use all words that have the "lang1" tags to build a character 5-grams LM for "lang1" and the same for "lang2". We apply all of the created CLM to each word in the training data to find their character sequence probabilities in each language in the training data. To increase the difference between the feature vectors of the words related to "lang1" and those related to "lang2", we use a word level unigram LM for one of the two languages in the training data. In practice, we pick the language where large corpora exist in order to build the LM. Then we apply the unigram LM to each word in the training data to find their word level probability. For the "ne" feature, we use the tagged named entities words from the training data as a lookup table. Then we put one in this feature if the word in the input tweet can be found in that lookup table, otherwise it is zero. We use SPLIT (Al-Badrashiny et al., 2016) to check if the word is numbers, dates, urls, emoticons, sounds, or punctuation. Then if the word is found to be any of these types, we put one the "is other" feature, otherwise it is zero.

3 Experimental Setup

Table 1 shows the labels distribution of each language in the training and dev sets. The lang1, lang2 labels refer to the two languages addressed in the dataset name, for example for the language pair English-Spanish, lang1 is English and lang2 is Spanish, in that order.

We also used the English Gigaword (LDC, 2003b) to build the unigram word level LM for the English part in English-Spanish. And the Arabic Gigaword (LDC, 2003a) to build the unigram word level LM for the Arabic part in MSA-DA.

4 Evaluation

Table 2 shows the best results we got on the dev sets of both language-pairs. The best results we got was by tuning the CRF classifier to use a window of 17 words (eight words before and after the current words).

	MSA-DA-Dev	EN-SP-Dev
lang1	81%	95%
lang2	83%	94%
mixed	0%	0%
ne	91%	70%
ambiguous	0%	0%
fw	0%	0%
other	99%	97%
unk	0%	12%
w-avg F-score	85%	94%

Table 2: Summary results of our system performance on the dev data of both language-pairs. For each group, the F-score is presented for all tags followed by the weighted average F-score for all tags.

Table 3 shows the results on the test set.

	MSA-DA-Test	EN-SP-Test
lang1	77%	81%
lang2	83%	95%
mixed	0%	0%
ne	83%	23%
ambiguous	0%	0%
fw	0%	0%
other	99%	95%
unk	0%	0%
w-avg F-score	83%	91%

Table 3: Summary results of our system performance on the test data of both language-pairs. For each group, the F-score is presented for all tags followed by the weighted average F-score for all tags.

The results show that the our system works better on the EN-SP data than the MSA-DA because, the words in the MSA and DA languages do not create disjoint sets, there is significant overlap hence they share significant character and word patterns. Hence, modeling more nuanced features is needed such as POS tags and morphological information to

improve the performance on the MSA-DA data. The main tag that need some more improvement is the "ne". It needs some other sophisticated techniques other than just using a lookup table. We also misunderstood the "others" tag in the Spanish-English data. We gave any word that starts with # the "other" label as in the Arabic guidelines, which affected our final results.

The main advantage of the proposed system is that it is language independent since it does not require any language-dependent components. Finally, the simplicity of our system made it very fast. It can process up to 20,000 words/sec; which renders it very efficient and amenable to large scale processing especially if a language identification module is required as a preprocessing step in some other applications (ex. Machine translation)

References

Mohamed Al-Badrashiny and Mona Diab. 2016. Lili: A simple language independent approach for language identification. In *The 26th International Conference on Computational Linguistics (COLING 2016). Osaka, Japan*.

Mohamed Al-Badrashiny, Ramy Eskander, Nizar Habash, and Owen Rambow. 2014. Automatic Transliteration of Romanized Dialectal Arabic. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning*, pages 30–38, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Mohamed Al-Badrashiny, Heba Elfardy, and Mona Diab. 2015. Aida2: A hybrid approach for token and sentence level dialect identification in arabic. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*, pages 42–51, Beijing, China, July. Association for Computational Linguistics.

Mohamed Al-Badrashiny, Arfath Pasha, Mona Diab, Nizar Habash, Owen Rambow, Wael Salloum, and Ramy Eskander. 2016. Split: Smart preprocessing (quasi) language independent tool. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Kfir Bar and Nachum Dershowitz. 2014. The tel aviv university system for the code-switching workshop shared task. In *Proceedings of the First Workshop on Computational Approaches to Code Switching*, pages 139–143, Doha, Qatar, October. Association for Computational Linguistics.

Utsab Barman, Joachim Wagner, Grzegorz Chrupała, and Jennifer Foster. 2014. Dcu-uvt: Word-level language classification with code-mixed data. In *Proceedings of the First Workshop on Computational Approaches to Code Switching*, pages 127–132, Doha, Qatar, October. Association for Computational Linguistics.

Ramy Eskander, Mohamed Al-Badrashiny, Nizar Habash, and Owen Rambow. 2014. Foreign words and the automatic processing of arabic social media text written in roman script. *In Proceedings of the First Workshop on Computational Approaches to Code-Switching. EMNLP 2014, Conference on Empirical Methods in Natural Language Processing, October, 2014, Doha, Qatar*.

LDC. 2003a. Arabic Gigaword Fifth Edition LDC2011T11. Linguistic Data Consortium.

LDC. 2003b. English Gigaword LDC2003T05. Linguistic Data Consortium.

Giovanni Molina, Fahad AlGhamdi, Mahmoud Ghoneim, Abdelati Hawwari, Nicolas Rey-Villamizar, Mona Diab, and Thamar Solorio. 2016. Overview for the second shared task on language identification in code-switched data. In *Proceedings of The EMNLP 2016 Second Workshop on Computational Approaches to Linguistic Code Switching (CALCS)*.

Arfath Pasha, Mohamed Al-Badrashiny, Mona Diab, Ahmed El Kholy, Ramy Eskander, Nizar Habash, Manoj Pooleery, Owen Rambow, and Ryan M. Roth. 2014. MADAMIRA: A Fast, Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic. In *Proceedings of LREC*, Reykjavik, Iceland.

Fei Sha and Fernando Pereira. 2003. Shallow parsing with conditional random fields. In *Proceedings of Human Language Technology-NAACL*, pages 213–220, Edmonton, Canada.

Andreas Stolcke. 2002. Srilm an extensible language modeling toolkit. In *Proceedings of the International Conference on Spoken Language Processing (ICSLP)*.