Code Switch Point Detection in Arabic

Heba Elfardy[1], Mohamed Al-Badrashiny[1], and Mona Diab[2]

[1] Columbia University
[2] The George Washington University
heba@cs.columbia.edu,
badrashiny@ccls.columbia.edu,
mtdiab@gwu.edu

Abstract. This paper introduces a dual-mode stochastic system to automatically identify linguistic code switch points in Arabic. The first of these modes determines the most likely word tag (i.e. dialect or modern standard Arabic) by choosing the sequence of Arabic word tags with maximum marginal probability via lattice search and 5-gram probability estimation. When words are out of vocabulary, the system switches to the second mode which uses a dialectal Arabic (DA) and modern standard Arabic (MSA) morphological analyzer. If the OOV word is analyzable using the DA morphological analyzer only, it is tagged as "DA", if it is analyzable using the "MSA" morphological analyzer only, it is tagged as MSA, otherwise if analyzable using both of them, then it is tagged as "both". The system yields an $F_{\beta=1}$ score of 76.9% on the development dataset and 76.5% on the held-out test dataset, both judged against human-annotated Egyptian forum data.

Keywords: Linguistic Code Switching, Diglossia, Language Modeling, Arabic, Dialectal Arabic Identification.

1 Introduction

Linguistic code switching (LCS) refers to the use of more than one language in the same conversation, either inter-utterance or intra-utterance. LCS is pervasively present in informal written genres such as social media. The phenomenon is even more pronounced in diglossic languages like Arabic in which two forms of the language co-exist. Identifying LCS in this case is more subtle in particular in the intra-utterance setting.[1] This paper aims to tackle the problem of code-switch point (CSP) detection in a given Arabic sentence. A language-modeling (LM) based approach is presented for the automatic identification of CSP in a hybrid text of modern standard Arabic (MSA) and Egyptian dialect (EDA) text. We examine the effect of varying the size of the LM as well as measuring the impact of using a morphological analyzer on the performance. The results are compared against our previous work [4]. The current system outperforms our previous implementation by a significant margin of an absolute 4.4% improvement, with an $F_{\beta=1}$ score of 76.5% compared to 72.1%.

[1] For a literature review, we direct the reader to our COLING 2012 paper [4].

E. Métais et al. (Eds.): NLDB 2013, LNCS 7934, pp. 412–416, 2013.

2 Approach

The hybrid system that is introduced here uses a LM with a back off to a morphological analyzer (MA) to handle out of vocabulary (OOV) words to automatically identify the CSP in Arabic utterances. While the MA approach achieves a far better coverage of the words in a highly derivative and inflective language such as Arabic, it is not able to take context into consideration. On the other hand, LMs yield better disambiguation results because they model context in the process.

2.1 Language Model

The system uses the MSA and EDA web-log corpora from the Linguistic Data Consortium (LDC) to build the language models. [2]Half of the tokens in the language model come from MSA corpora while the other half come from EDA corpora. The prior probabilities of each MSA and EDA word are calculated based on their frequency in the MSA and DA corpora, respectively. For example, the EDA word *ktyr*,[3] meaning *much*, will have a probability of 0 for being tagged as *MSA* since it would not occur in the MSA corpora, and a probability of 1 for being tagged as *EDA*. Other words can have different probabilities depending on their unigram frequencies in both corpora.

All tokens in the MSA corpora are then tagged as *MSA* and all those in the EDA corpora as *EDA*. Using SRILM [7] and the tagged datasets, a 5-gram LM is built with a modified Kneser-Ney discounting.

The LM and the prior probabilities are used as inputs to SRILM's *disambig* utility which uses them on a given untagged sentence to perform a lattice search and return the best sequence of tags for the given sentence.

2.2 Morphological Analyzer

All OOVs are run through CALIMA [5], an MSA and EDA morphological analyzer based on the both the SAMA [6] MA and database as well as the Tharwa three way MSA-EDA-ENG dictionary [2]. CALIMA returns all MSA and EDA analyses for a given word. The OOV word is tagged as "both" if it has MSA and EDA analyses. While it is tagged as "MSA" or "EDA" if it has only MSA or EDA analyses, respectively.

3 Evaluation Dataset

We use three different sources of web-log data to create our evaluation dataset. The first of which comes from the Arabic Online Commentary dataset that was

[2] The LDC numbers of these corpora are 2006{E39, E44, E94, G05, G09, G10}, 2008{E42, E61, E62, G05}, 2009{E08, E108, E114, E72, G01}, 2010{T17, T21, T23}, 2011{T03}, 2012{E107, E19, E30, E51, E54, E75, E89, E94, E98, E99}.

[3] We use Buckwalter transliteration scheme, `http://www.qamus.org/transliteration.htm`

produced by [8] and consists of user commentaries from an Egyptian newspaper while the second one was crawled from Egyptian discussion forums for the COLABA project [1] and finally the third one comes from one of the LDC corpora that are used to build the EDA language model. All datasets are manually annotated by a trained linguist using a variant of the guidelines that are described in [3]. In this variant of the guidelines, the annotation is purely contextual, so that if a word is used with the same sense in MSA and EDA, its label is determined based on the context it occurs in. In rare cases, where enough context is not present, a *both* class is used indicating that the word could be both MSA and EDA. Since we are not currently targeting romanized-text and named-entity identification, we exclude all entries that are labeled as Foreign, or Named-Entity from our evaluation, which correspond to a total of 8.4% of our dataset. Moreover, we also exclude unknown words and typos which only represent 0.7% of our dataset. We split our dataset into a development set for tuning and a held-out set for testing. The development-set has 19,954 MSA tokens (7,748 types), 9,771 EDA tokens (4,379 types) and 9 Both tokens (9 types). The test-set comprises 15,462 MSA tokens (6,887 types), 16,242 EDA tokens (6,151 types) and 5 Both tokens (5 types).

4 Experimental Results

We investigate two experimental conditions: one with the morphological analyzer as a back off turned on, the second mode has the morphological analyzer turned off. Both conditions experiment with varying the size of the LM as follows: 2, 4, 8, 16, 28M words, respectively. We employ two baselines: MAJB, a majority baseline that tags all words with the most frequent tag in our data set; the second baseline, COLB, is the approach presented in [4] using the same datasets that we used in building our language models. Figure 1 shows the $F_{\beta=1}$ of both sets of experiments against the baselines. Our approach significantly outperforms both baselines. One surprising observation is that the $F_{\beta=1}$ decreases as the size of the LM increases beyond 4 million tokens (with a slight drop at the 8M mark). We surmise that this is because as the size of the language model increases, the shared ngrams between MSA and EDA increases. For example, for the 4M LM (where we note the highest $F_{\beta=1}$ score), the shared types represent 21.2% while for the largest LM of size 28M, the shared types represent 27.6% . This causes more confusability between the classes for larger LMs which explains the lower $F_{\beta=1}$ scores despite the higher coverage.

As expected backing-off to the morphological analyzer improves the results especially for the smaller LMs where there is less coverage. However as the size of the LM increases, the coverage increases and the percentage of OOV decreases hence the morphological analyzer becomes less useful. For example, the percentage of OOVs for the 4M LM (when not backing-off to the morphological analyzer) is 7.2% while for the 28M LM it is 3.1%.

On the test set, the system outperforms both baselines with an $F_{\beta=1}$ score of 76.5% using the best configuration (4M tokens with back off to the morphological

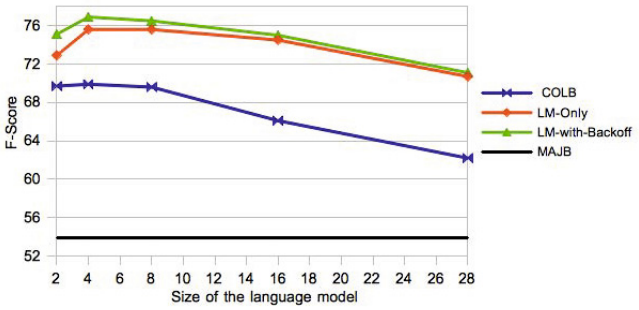

Fig. 1. Weighted Average of F-Scores of the MSA and DA classes with different experimental setups against the baseline systems, MAJB and COLB

analyzer) compared to 34.7% for the majority baseline MAJB and 72.1% for our high baseline system, COLB.

5 Conclusion

We presented a new dual-mode stochastic system to automatically perform point-level identification of linguistic code switches in Arabic. We studied the impact of varying the size of the language model with and without employing a morphological analyzer as a back-off method to handle the OOV. Our best (using the LM plus the morphological analyzer as a back-off) system achieves an F-Score of 76.9% and 76.5% on the development and test datasets, respectively. These results outperformed both the majority baseline and our previous approach introduced in [4].

Acknowledgments. This work is supported by the Defense Advanced Research Projects Agency (DARPA) BOLT program under contract number HR0011-12-C-0014.

References

1. Diab, M., Habash, N., Rambow, O., Altantawy, M., Benajiba, Y.: Colaba: Arabic dialect annotation and processing. In: LREC Workshop on Semitic Language Processing, pp. 66–74 (2010)
2. Diab, M., Hawwari, A., Elfardy, H., Dasigi, P., Al-Badrashiny, M., Eskandar, R., Habash, N.: Tharwa: A multi-dialectal multi-lingual machine readable dictionary (forthcoming, 2013)
3. Elfardy, H., Diab, M.: Simplified guidelines for the creation of large scale dialectal arabic annotations. In: LREC, Istanbul, Turkey (2012)

4. Elfardy, H., Diab, M.: Token level identification of linguistic code switching. In: COLING, Mumbai, India (2012)
5. Habash, N., Eskander, R., Hawwari, A.: A Morphological Analyzer for Egyptian Arabic. In: NAACL-HLT Workshop on Computational Morphology and Phonology (2012)
6. Maamouri, M., Graff, D., Bouziri, B., Krouna, S., Bies, A., Kulick, S.: Ldc standard arabic morphological analyzer (sama) version 3.1 (2010)
7. Stolcke, A.: Srilm an extensible language modeling toolkit. In: ICSLP (2002)
8. Zaidan, O.F., Callison-Burch, C.: The arabic online commentary dataset: an annotated dataset of informal arabic with high dialectal content. In: ACL (2011)