INTERSPEECH 2019
September 15–19, 2019, Graz, Austria

Improving Code-Switched Language Modeling Performance Using Cognate Features

Victor Soto, Julia Hirschberg

Columbia University, Computer Science Department
New York, NY, 10027
vsoto@cs.columbia.edu, julia@cs.columbia.edu

Abstract

We have found that *cognate words*, defined as sets of words used in multiple languages that share a common etymology, can in fact elicit code-switching or language mixing between the languages. This paper focuses on how information about cognate words can improve language modeling performance of code-switched English-Spanish (EN-ES) language. We have found that the degree of semantic, phonetic or lexical overlap between a pair of cognate words is a useful feature in identifying code-switching in language. We derive a set of spelling, phonetic and semantic features from a list of of EN-ES cognates and run experiments on a corpus of conversational code-switched EN-ES. First, we show that there exists a strong statistical relationship between these cognate-based features and code-switching in the corpus. Secondly, we demonstrate that language models using these features obtain similar performance improvements as do other manually tagged features including language and part-of-speech tags. We conclude that cognate features can be a useful set of automatically-derived features that can be easily obtained for any pair of languages.

Index Terms: language modeling, code-switching, cognates

1. Introduction

Code-switching (CS) is the alternate use of two or more languages during communication. CS can be intra-sentential, if it happens within the boundaries of a sentence or utterance (e.g. "I love baloncesto"), or inter-sentential otherwise (e.g. "I'm leaving. Adios."). In Natural Language Processing and Speech Analysis, intra-sentential CS is of particular concern because it often renders monolingual parsing, part-of-speech (POS) tagging, machine translation, summarization, and speech recognition systems, among others, useless, since the language being analyzed at any point in the process is unknown.

The most difficult challenge to developing new models for identifying code-switched data is the lack of manually-annotated resources for most language pairs, and hence, the lack of knowledge of how and when CS is likely to occur. In particular, for the task of Language Modeling (LM), which consists of predicting the next word given a sequence of words, the question of how a code-switch is triggered becomes particularly important. While some linguistics literature on CS has proposed that a) cognates, defined as words in two different languages with the same etymology and similar spelling and meaning, are more likely to precede a code-switch, and that b) there are syntactic constraints to CS, there has been little research validating these proposals empirically. On the NLP field, there has been some prior research on the tasks of language modeling for CS data using manually labeled language identification and syntactic information. These features, while useful, are difficult to obtain both in terms of expense and in the difficulty of training annotators. In this paper, we propose a new set of spelling, pronunciation and semantic features extracted from automatically-extracted lists of cognate words, and compare their performance to hand-labeled features. We find that the new set of cognate-based features we propose does indeed add similar improvements to our LMs compared to manually-labeled Language ID (LID) tags and POS tags and are much easier to obtain. Better LMs for code-switched data can thus be developed without the need for large amounts of manually-labeled training data, thus leading to improvements in speech and language processing of CS in many more language pairs.

This paper is organized as follows. Section 2 provides an overview of previous work on CS for LM and on cognate words and their role in CS. Section 3 describes the corpus used in this paper. Section 4 outlines the cognate-based features we are proposing. Section 5 gives a short introduction to the Factored Language Model (FLM) approach we are using for our experiments. Section 6 describes our experiments and Section 7 presents our conclusions and plans for future research.

2. Previous Work

In the last decade there has been increasing interest in tackling the problem of modeling of code-switched language in the computational linguistics community. Most efforts have focused on applying machine learning methods to the task of language modeling. The first example of a statistical language model (SLM) applied to CS data was presented in [1], where the authors trained 2-,3-,4- and 5-grams on a very small corpus (unlabeled for language ID) obtaining perplexity values ranging from 49.40 to 50.95. [2] is the first example of an SLM to incorporate a syntactical constraint ("the equivalence constraint" [3] which states that "the order of constituents immediately adjacent to the code-switching point must be the same in both language's grammars") from the linguistics community. This work achieved a word error rate of 35.2% and 45.9% in two conversational speech corpora. In [4] the same authors incorporated the Functional Head Constraint [5] (which states that code-switching cannot occur between a functional head and its complement) and achieved further improvements in word error rates of 33.70% and 43.58% on the same corpora.

[6] performed LM experiments using FLMs and RNNs on the SEAME corpus of English and Mandarin code-switching. They found that the RNNs achieved better results than FLMs and demonstrated that LID and POS tags are useful features for CS LM. However, their perplexity values were very high (239.21 for the best single model and 192.08 for the best combined model). In a similar vein as the work we present here, [7] presented an analysis that shows that certain words and POS tags are more likely to precede a code-switch; however their proposed RNN model for LM ended up using only POS classes

Copyright © 2019 ISCA

http://dx.doi.org/10.21437/Interspeech.2019-2681

Table 1: *Number of CS and monolingual utterances split by utterances that contain a cognate or not.*

		Cognate	
		no	yes
CS	no	20,029	18,767
	yes	1,037	1,937

Table 2: *Number of CS and non-CS words split by cognate and non cognate words.*

		Cognate	
		no	yes
CS	no	222,703	34,877
	yes	3,740	453

and words as input, without any attempt to flag what *type* of words were useful. [8] proposed that a multi-task learning approach to POS tagging and LM can help improve LM performance and showed relative perplexity improvements of 9.7% on the SEAME corpus. Similarly, [9] achieved some improvements on the joint task of LID tagging and LM.

Work on cognates and their role in triggering CS by the Australian linguist Michael Clyne [10, 11] [12], an expert on bilingualism, proposed that cognate words can facilitate CS if they occur immediately preceding or following a CS. A small statistical analysis study in [13] confirmed that cognate words do trigger a subsequent CS. In [14], we tested Clyne's hypothesis on the Miami Bangor corpus and proved with high statistical significance that a) cognates and code-switches do tend to occur in the same utterance; b) cognates are very likely to immediately precede a code-switch, and c) there is a strong statistical relationship between code-switching and part-of-speech tags that immediately precede or occur after a code-switch. In this paper, we continue to investigate this relationship using cognate-based features, POS tags and LID tags as input for the LM task.

3. The Miami Bangor Corpus

The Miami Bangor corpus is a conversational speech corpus recorded from bilingual Spanish-English speakers living in Miami, FL. It includes 56 files of conversational speech from 84 speakers. The corpus consists of 242,475 transcribed words (333,069 tokens included punctuation) and 35 hours of recorded conversation. The manual transcripts include beginning and end times of utterances and per word LID. Each token is tagged with an LID label as English, Spanish, Ambiguous, Mixed, Punctuation or Other. 53.48% of transcribed words are English, 27.28% Spanish, although this distribution is different for the subset of CS utterances (38.98% and 46.12% tokens in English and Spanish respectively). In [15], we crowdsourced part-of-speech tags using the Universal POS Tagset [16] obtaining high inter-annotator agreement (0.95).

The corpus, as normalized by us [17], has a total of 42,910 utterances of which 2,974 contain at least one code-switch (7.12%) and 20,704 contain at least a cognate (49.57%). At the word level, the corpus has a total of 4,193 code-switched words (1.6%) and 35,330 cognate words (13.5%). The corpus can be obtained from GitHub[1]. Tables 1 and 2 show contingency tables of the cognate and CS distribution across the corpus at the utterance and word level respectively.

This dataset was split into train, development and test sets for the experiments presented in Section 6. The size of each split is detailed in Table 3.

Table 3: *Number of sentences and tokens in the full Miami Bangor Corpus and each of its splits.*

Split	# Sents	# Toks
Full	42.9K	321,630
Train	36,710	274,863
Dev	2,000	15,588
Test	4,200	31,179

4. Feature Engineering

4.1. Feature Extraction

In this paper we use the list $\mathcal{L} = \{(e^k, s^k)\}$ of English-Spanish pairs of cognate words described in [14], which can be obtained from Github[2]. Each entry in the list consists of an English word e^k and a Spanish word s^k of the same cognate (e.g. "mathematics" and "matemáticas"). The list has a total of 3,423 cognate word pairs, of which a total of 1,305 appear at least once in the MB corpus. For each of these word pairs (e^k, s^k), we extract the following set of features, $f_l^k = f_l(e^k, s^k)$ quantifying the difference between cognate pairs in terms of spelling, pronunciation, and meaning:

Spelling features: To compute these features we measured the distance or similarity between the sequence of letters of the pair of cognates. Distances used include the Damerau–Levenshtein (DLD), Hamming (HD), and Levenshtein (LD) distance. We also computed the Jaro (JS) and Jaro-Winkler (JWS) similarities. We also include a 'perfect cognate' feature which is 1 if the spelling is identical in Spanish and English (not accounting for tildes) and 0 otherwise. For example for the cognate pairs "mathematics" and "matemáticas", the Levenshtein distance is 0.18.

Pronunciation features: These features reflect how different the pronunciation between the pair of words is. We used the CMU English pronunciation dictionary and Spanish pronunciation dictionary and trained a grapheme-to-phoneme system using the CMU Sphinx sequence-to-sequence system described in [18]. Once all the pronunciations were obtained, we computed the distance between both pronunciations using the Binary (BD), Hamming (HD), Jaccard (JD) and Levenshtein (LD) distances.

Semantic features: These features are intended to reflect how close in meaning the two words in each cognate pair are. We used the MUSE bilingual word embeddings [19] and computed the Euclidean (EUC) distance and the Cosine (COS) Similarity between the cognate pairs. Only 15 cognate words that appeared in the MB corpus were not covered by the bilingual embeddings.

4.2. Feature Normalization

All the features not naturally bounded to $[0, 1]$ were normalized by the feature's maximum possible value, which for most distances is the maximum sequence length of one of the cognates in the pair. All the distance features were transformed into similarities using a simple transformation $sim = 1 - dist$.

[1] https://github.com/vsoto/crowdsourced_bangor

[2] https://github.com/vsoto/cognates_en_es

4.3. Statistical relationship between Code-switching and Cognate Features

To analyze the relationship between CS and the cognate-based features, and to determine if the features can be predictive of code-switching behavior, we first looked at how similar the distribution of these features is when looking at the words surrounding a code-switch and the rest of the utterance. To do so, we ran the Kruskal-Wallis statistical test to compare the distribution of features with respect to their position relative to a (labeled) code-switch. Kruskal-Wallis tests the null hypothesis that the population medians for two or more groups are equal, which can be rejected with a sufficiently small p-value. If the distributions (medians) of two subgroups of feature values are different enough, these features will be potentially usable for code-switch detection and language modeling.

To run the statistical significance tests we assign feature values f_l to every word w_i in an utterance: If the word w_i is a cognate (e^k, s^k) present in the list of cognates, the word is given the feature value $f_l(w_i) = f_l(e^k, s^k)$; otherwise, the word is assigned the minimum possible value for that feature, which is zero. For example, for the phrase "very simpático", where "very" is not a cognate and "simpático" is a cognate in the list, we would assign a zero to the first word and the pertinent feature value to the second word. We run the statistical test for each feature described in Section 4.1 and in three different modalities to compare the feature distributions of a) code-switched words and the rest of words in an utterance; b) words that immediately precede a code-switch and the rest of words in an utterance and c) words that immediately follow a CS and the rest of the words in the utterance.

Table 4: *Statistical significance results of running the Kruskal-Wallis test by ranks of all the features split into two groups. Three pairs of groups are tested: words preceding a code-switch and the rest of words, code-switched words and the rest of words; and words following a code-switch and the rest. Check marks ✓ indicate that there is a statistically significant difference between the distribution of the features values of the two groups.*

Group	Feat.	Prec	CS	After
Spelling	DL	✓	✓	-
	Hamming	✓	✓	-
	Jaro	✓	✓	-
	Jaro-Winkler	✓	✓	-
	Levenshtein	✓	✓	-
	Perfect	✓	✓	-
Pron.	Binary	-	✓	-
	Hamming	✓	✓	-
	Jaccard	-	✓	-
	Levenshtein	✓	✓	-
Semantic	Cosine	✓	✓	-
	Euclidean	✓	✓	-

Results of these tests are presented in Table 4. In this table, each row contains the results from the Kruskal-Wallis test for a given feature and each column specifies the distributions that are being compared. Column 3 compares the feature distributions of the words immediately preceding a code-switch and the rest of the words in the corpus. Column 4 compares the feature distributions of the code-switched words and the rest of the words in the corpus; and column 5 compares the feature distributions of the words immediately following a code-switch and the rest of the words in the corpus. Check marks indicate p-values $p < 0.001$.

Following the same trend that we observed in [14], the p-values confirm that all engineered features values have different median values when they **precede** a code-switch and when they **are** code-switched; however, they do not present statistical differences when they immediately **follow** a code-switch.

For the spelling features, all show significantly different distributions when the word they are calculated from precedes ($10^{-19} < p < 10^{-15}$) or is itself a code-switch ($10^{-20} < p < 10^{-8}$). Similarly for the pronunciation features, p-values range from $10^{-22} < p < 10^{-10}$ for feature values of code-switched words and $10^{-18} < p < 10^{-15}$ for feature values for words immediately preceding a CS (Hamming and Levenshtein distance). Tests run on semantic features return smaller p-values when focused on words preceding a switch ($10^{-7} < p < 10^{-4}$) but similar power on CS words ($10^{-22} < p < 10^{-20}$). Overall, the largest differences were always found on the CS word (perfect spelling, binary distance on pronunciation entries and cosine similarity on word embeddings).

5. Factored Language Models

Factored Language Models (FLMs) [20] are language models that encode each word w_i in a sentence as a vector of k factors $w_i = (f_i^1, \ldots, f_i^k) = f_t^{1:k} = F_i$, where each factor can be a feature of the word, i.e. the language of the word or its part-of-speech tag. An FLM is a directed graphical model where $p(F_t|F_{t-l}, \ldots, F_{t-1})$ can be factored into probabilities of the form $p(f|f_1, \ldots, f_N)$. An FLM is described by its backoff graph, which shows the various backoff paths from the parent node $p(F|F_1, \ldots, F_N)$ to the child node $P(F)$. Given a chosen backoff graph topology, FLMs can be trained using the Generalized Parallel Backoff algorithm, which allows the language model to back off on a single path or on multiple parallel paths simultaneously during runtime.

For the experiments presented in this paper, we used the FLM implementation in the SRILM toolkit [21, 22], which allows for fast training and evaluation of FLMs. Some of the key implementation issues when using FLMs are the choice of factors to use in the model and the design of the backoff graph. Many factors go into the design of the backoff graph, including: the topology of the graph (including the number of backoff graph nodes, and the dependencies between them) and the discounting, smoothing and combination options for each node. Given all these design factors, finding the optimal FLM structure for a given corpus is a highly intractable problem. We use GA-FLM [23], a genetic algorithm that searches over the space of possible FLMs structures optimizing for development set perplexity. Specifically, for each of FLMs trained on the next section, the GA-FLM was run on 10 generations, each one with a population size of 100, with a cross-over probability of 0.9 and a mutation probability of 0.01.

6. Experiments & Results

We start by training FLMs using exclusively word tokens and the gold features we have on the MB corpus: LID and POS tags. All FLMs are trained using features of two previous words (3-grams). Table 5 shows the perplexity achieved by the baseline tri-gram language models and by the same language models when adding the gold LID and POS features to the Bangor Corpus. The addition of the LID and POS tags separately help achieve similar improvements, from 73.57 down to 68.88 and 68.87 respectively. When used together the perplexity drops much further to 59.28, proving that the two features are com-

Table 5: *Test set perplexity of FLMs trained on word trigrams and LID and POS tags.*

Model	PP
W	73.57
W + LID	68.88
W + POS	68.87
W + LID + POS	59.28

plementary and equally useful for language modeling.

Table 6 shows the performance of a trigram FLM when adding the cognate-based features. The top subtable shows the LM performance when adding the cognate and perfect cognate flags. In both cases perplexity improves with respect to the 73.57 baseline, but none of the features are as useful as LID or POS tags for LM. The next three subtables show the perplexity of the LM when adding just one of the spelling, pronunciation, or semantic cognate-based features. For the lexical features, the best performance is achieved when using the Jaro-Winkler distance between the English and Spanish cognates (65.35); for the pronunciation features, the best performance is achieved when using the Hamming distance (65.99); and for the semantic features, both the cosine and euclidean distances perform similarly (66.02). Comparing tables 5 and 6, the cognate-based features can achieve better perplexity performance that the LID and POS tags features when used separately. This is important because LID and POS tags for this corpus were crowdsourced and expensive to obtain. However, no cognate-based feature helps achieve similar performance as the combination of the manual LID and POS tags.

Table 6: *Test set perplexity of FLMs trained on word trigrams and each of the cognate-based features.*

Model	PP
W + Cognate	70.17
W + Perfect Cognates	71.71
W + LEX(JWS)	**65.35**
W + LEX(LD)	65.88
W + LEX(DLD)	66.02
W + LEX(JS)	67.02
W + LEX(HD)	72.01
W + PRON(JD)	66.42
W + PRON(HD)	**65.99**
W + PRON(BD)	70.14
W + PRON(LD)	66.42
W + SEM(EUC)	66.02
W + SEM(COS)	66.02

Table 7 shows the perplexity performance of the FLM models when adding a combination of the cognate-based features. For each category (LEX, PRON and SEM) the best performing feature from Table 6 was chosen. The table shows that the combination of PRON+SEM, and the combination of LEX+PRON+SEM helps improve the perplexity achieved by the models shown in Table 6, although the gains are very small. We hypothesize that the combination of LEX and PRON features may not offer perplexity gains since the features are computed very similarly (the first as the string distance and the second as the distance between two phone sequences) whereas adding the SEM feature always helps improve performance. However, the addition of all cognate-based features does not bring performance improvements comparable to the addition of LID and POS tags (64.51 compared to 59.28).

Table 7: *Test set perplexity of FLMs using a combination of two or the three cognate-based features.*

Model	PP
W + LEX + PRON	66.23
W + LEX + SEM	65.90
W + PRON + SEM	64.95
W + LEX + PRON + SEM	64.51

We conclude the experiments by examining how much gain we can obtain from adding the cognate-based features to the LID and POS tags, which obtained a perplexity of 59.28 (see table 5). We see that adding any subset of cognate features adds value to the W+L+P model, with perplexity numbers ranging from 58.30 to 59.17, although these improvements are very small.

Table 8: *Test set perplexity of FLMs using cognate flags, LID and POS tags plus one set of one, two, or three cognate-based features.*

Model	PP
W + C + L + P	58.85
W + C + L + P + PRON	58.32
W + C + L + P + SEM	58.32
W + C + L + P + LEX	58.75
W + C + L + P + LEX + PRON	58.30
W + C + L + P + LEX + SEM	59.17
W + C + L + P + PRON + SEM	60.01
W + C + L + P + LEX + PRON + SEM	58.84

7. Conclusions

In this paper, we have proposed a new set of features extracted from lists of cognate words to improve CS detection. This set of features describes the semantic, orthographic and phonetic similarities across pairs of cognate words in English and Spanish. We first showed that there is a very high statistical relationship between these features and CS, which signals their potential usefulness for CS language modeling. We then showed that FLMs trained on these features achieve similar performance as FLMs trained on gold features like LID and POS tags separately. The three feature sets (semantic, orthographic and phonetic) do not appear to be very complementary and underperform when compared to the joint use of LID and POS tags, however they are much simpler and less expensive to obtain.

For future work, we plan to develop apply our newly developed language models on automatic speech recognizers for code-switched data.

8. References

[1] J. C. Franco and T. Solorio, "Baby-steps towards building a Spanglish language model," in *Proc. of International Conference on Intelligent Text Processing and Computational Linguistics*. Springer, 2007, pp. 75–84.

[2] Y. Li and P. Fung, "Code-switch language model with inversion constraints for mixed language speech recognition," in *Proc. of COLING*, 2012, pp. 1671—-1680.

[3] D. Sankoff and S. Poplack, "A formal grammar for code-switching," *Research on Language & Social Interaction*, vol. 14, no. 1, pp. 3–45, 1981.

[4] Y. Li and P. Fung, "Language modeling with functional head constraint for code switching speech recognition." in *Proc. of EMNLP*, 2014, pp. 907–916.

[5] H. M. Belazi, E. J. Rubin, and A. J. Toribio, "Code switching and x-bar theory: The functional head constraint," *Linguistic inquiry*, pp. 221–237, 1994.

[6] H. Adel, N. T. Vu, and T. Schultz, "Combination of recurrent neural networks and factored language models for code-switching language modeling." in *Proc. of ACL*, 2013, pp. 206–211.

[7] H. Adel, N. T. Vu, F. Kraus, T. Schlippe, H. Li, and T. Schultz, "Recurrent neural network language modeling for code switching conversational speech," in *Proc. of ICASSP*. IEEE, 2013, pp. 8411–8415.

[8] G. I. Winata, A. Madotto, C.-S. Wu, and P. Fung, "Code-switching language modeling using syntax-aware multi-task learning," in *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*. Association for Computational Linguistics, 2018, pp. 62–67. [Online]. Available: http://aclweb.org/anthology/W18-3207

[9] K. Chandu, T. Manzini, S. Singh, and A. W. Black, "Language informed modeling of code-switched text," in *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*. Association for Computational Linguistics, 2018, pp. 92–97. [Online]. Available: http://aclweb.org/anthology/W18-3211

[10] M. G. Clyne, *Transference and triggering: Observations on the language assimilation of postwar German-speaking migrants in Australia*. Martinus Nijhoff, 1967.

[11] ——, "Triggering and language processing." *Canadian Journal of Psychology/Revue canadienne de psychologie*, vol. 34, no. 4, p. 400, 1980.

[12] ——, *Dynamics of language contact: English and immigrant languages*. Cambridge University Press, 2003.

[13] M. Broersma and K. De Bot, "Triggered codeswitching: A corpus-based evaluation of the original triggering hypothesis and a new alternative," *Bilingualism: Language and cognition*, vol. 9, no. 1, pp. 1–13, 2006.

[14] V. Soto, N. Cestero, and J. Hirschberg, "The role of cognate words, POS tags, and entrainment in code-switching." in *Proc. of Interspeech*, 2018, pp. 1938–1942. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2018-1099

[15] V. Soto and J. Hirschberg, "Crowdsourcing universal part-of-speech tags for code-switching," in *Interspeech*, 2017, pp. 77–81.

[16] S. Petrov, D. Das, and R. McDonald, "A universal part-of-speech tagset," *arXiv preprint arXiv:1104.2086*, 2011.

[17] V. Soto and J. Hirschberg, "Joint part-of-speech and language ID tagging for code-switched data," in *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*. Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 1–10. [Online]. Available: https://www.aclweb.org/anthology/W18-3201

[18] K. Yao and G. Zweig, "Sequence-to-sequence neural net models for grapheme-to-phoneme conversion," *CoRR*, vol. abs/1506.00196, 2015. [Online]. Available: http://arxiv.org/abs/1506.00196

[19] A. Conneau, G. Lample, M. Ranzato, L. Denoyer, and H. Jégou, "Word translation without parallel data," *arXiv preprint arXiv:1710.04087*, 2017.

[20] J. A. Bilmes and K. Kirchhoff, "Factored language models and generalized parallel backoff," in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology: companion volume of the Proceedings of HLT-NAACL 2003–short papers-Volume 2*. Association for Computational Linguistics, 2003, pp. 4–6.

[21] A. Stolcke, "Srilm-an extensible language modeling toolkit," in *Seventh international conference on spoken language processing*, 2002.

[22] A. Stolcke, J. Zheng, W. Wang, and V. Abrash, "Srilm at sixteen: Update and outlook," in *Proceedings of IEEE automatic speech recognition and understanding workshop*, vol. 5, 2011.

[23] K. Duh and K. Kirchhoff, "Automatic learning of language model structure," in *Proceedings of the 20th international conference on Computational Linguistics*. Association for Computational Linguistics, 2004, p. 148.