Token Level Code-Switching Detection Using Wikipedia as a Lexical Resource

Daniel Claeser(✉), Dennis Felske(✉), and Samantha Kent(✉)

Fraunhofer FKIE, Fraunhoferstraße 20, 53343 Wachtberg, Germany
{daniel.claeser,dennis.felske,samantha.kent}@fkie.fraunhofer.de

Abstract. We present a novel lexicon-based classification approach for code-switching detection on Twitter. The main aim is to develop a simple lexical look-up classifier based on frequency information retrieved from Wikipedia. We evaluate the classifier using three different language pairs: Spanish-English, Dutch-English, and German-Turkish. The results indicate that our figures for Spanish-English are competitive with current state of the art classifiers, even though the approach is simplistic and based solely on word frequency information.

1 Introduction

Code-switching (CS) or code-mixing can be defined as a linguistic phenomenon in which multilingual speakers use languages interchangeably. A distinction is made between inter-sentential CS, where the switch occurs at sentence level, and intra-sentential CS, where the switch occurs within a sentence at the phrase or word level (Bullock and Toribio 2009). In turn, intra-sentential CS can be divided into two different types. Alternation is the switching of different languages whilst keeping the grammatical structure of each language intact. Contrastingly, in insertion, lexical items from one language are included within the grammatical structure of another (Muysken 2000).

In recent years, multilingual written communication that includes these different types of CS has become more prevalent and there has been a growing interest in the automatic identification of codeswitched language on social media. This paper seeks to contribute to that growing body of work and introduce a simple lexical look-up classifier that identifies code-switching between different languages on Twitter. The focus lies on three different language pairs: Spanish-English, Dutch-English, and German-Turkish.

2 Related Work

Currently, a range of different classification approaches have been presented for a variety of different languages (see Solorio et al. (2014) and Molino et al. (2016) for a current overview). Methods vary from the use of more complex deep learning algorithms (Jaech et al. 2016) to a range of different lexicon-based approaches, those of which are most relevant to our approach are discussed in turn below.

G. Rehm and T. Declerck (Eds.): GSCL 2017, LNAI 10713, pp. 192–198, 2018.
https://doi.org/10.1007/978-3-319-73706-5_16

Maharjan et al. (2015) compare a lexical lookup classification approach to current state of the art classifiers in order to identify Spanish-English code-switched Tweets. They demonstrate a very simple dictionary approach in which the classifier assigns a language based on whether the token was present in a Spanish or English dictionary. If the token is either present in both dictionaries or absent in both dictionaries, the tag associated with the majority language in the training data is assigned. The results illustrate that the most elementary binary classification approach yields an F1 score of 0.61 at Tweet level and 0.73 at token level.

Chanda et al. (2016) combine a dictionary method with the use of n-gram categorization and additional processing of Bengali suffixes in order to identify English-Bengali CS. In contrast to the previous binary classification approach, the lexical look-up in the English, Bengali, and hand-crafted slang dictionary does not solely determine the language of the token. The inclusion of the additional features yields an accuracy level of 86.27%. Notably, when predicting the languages based solely on the dictionary approach an accuracy level similar to the approach outlined above is achieved (72.54%). In a further step, the authors compare various machine learning techniques to construct the actual classifier and find that the most accurate results were achieved using an IBk algorithm, where an accuracy of 90.54% is reached in their social media corpus (compared to 91.65% in their corpus not extracted from social media).

Shirvani et al. (2016) combine 14 different features, including character n-grams, prefixes and suffixes, a Spanish-English dictionary, Spanish and English Part-Of-Speech (POS) tagging, Brown clustering, as well as a number of additional binary features. Logistic regression is used to determine the probability of the various possible labels using different combinations of these 14 features. This language classifier is more complex and contains more features compared to the previous approach. The results indicate a further increase in overall performance with a weighted F1 score of 91.3% at Tweet level and an overall accuracy of 97.3% at token level, with 93.8% and 98.4% for English and Spanish respectively.

3 Datasets

We used *corpora* that were previously collected and annotated to evaluate the classifier. The English-Spanish Twitter corpus was provided for the Shared Task Challenge for EMNLP 2016 (Molina et al. 2016), the Turkish-German corpus was created by Çetinoğlu (2016) and the Dutch-English Twitter corpus was provided by the University of Amsterdam (Dongen 2017). The first two corpora are distributed in the form of Tweet IDs that are to be downloaded using the Twitter API. However, due to Twitter's policy, we were only able to download a fraction of the original corpora, even though both were assembled in 2016. Overall, we managed to procure a total of 1028 Tweets (7133 tokens) from the English-Spanish corpus and 145 Tweets (1720 tokens) from Turkish-German corpus. The Dutch-English corpus was provided in plain text and we thus managed to reproduce it completely (1284 Tweets, 16050 tokens). For each of the language

pairs, we were able to use the full Twitter corpus as an evaluation set because we did not need a training corpus.

The *dictionaries* were built using the Wikipedia dumps for each of the five languages in the corpora (version: "all pages with complete edit history" on 01/03/2017). Crucially, those packages contain both the user discussion sections for each article as well as the actual article itself. The input size of the dictionaries varies for each of the languages, the influence of which will be discussed in the results section. The dictionaries were created as follows. The basic format is a token list which was obtained by parsing the Wikipedia dumps for the respective language. The raw input was stripped of all special characters before being changed to lower case, tokenised and ranked according to their frequency. Later, the dictionaries were cropped at 5 million types each, based on the idea that tokens that rank lower than 5 million are mostly hapax legomena.

Using Wikipedia as the input source to create the dictionaries has a number of different advantages. Firstly, it is freely available and distributable under the CC license, free of charge, and easy to access for any language that is present on Wikipedia. Secondly, due to the fact that the dictionaries contain text from both the articles and the comments section, we captured both formal and informal language. Consequently, the dictionaries contain a wide range of vocabulary, ranging from subject specific vocabulary to a variety of different abbreviations. This turned out to be a crucial aspect, because almost all tokens in the Tweets are present in the dictionaries even though language on social media is characteristically colloquial. Furthermore, Wikipedia tends to reflect current topics, even if there is a delay compared to social media, which ensures that the vocabulary in the dictionaries is up to date.

4 Classification

The classification process is based on a number of different assumptions. Firstly, we assume that if the dictionaries are large enough, all tokens will be present in all dictionaries, regardless of the language. Crucially however, the rank of the token will be different in each of the dictionaries, and it is likely that a word stems from the language in which the rank in the dictionary is the highest. So in the first step of the token-level classification of Tweets, the rank of the token is retrieved from the respective language dictionaries, and the language in which the rank is highest is assigned to the token. In the rare case that a token is in fact not present in the dictionaries the tag 'none' is assigned. In the final step, all tokens that are classified as 'none' are assigned to the majority language of the Tweet.

Secondly, the assumption is made that some tokens are not unique to a specific language. This is particularly true for language pairs such as Dutch-English, which share many overlapping lexical items. So in order to account for orthographically identical words that are frequently used in both languages, two further steps are introduced. In the first of these steps, tokens that are ranked very highly in both languages simultaneously, for example the word 'me'

in English and Spanish, are considered to be grammatical function words that are identical in both languages and thus should be assigned to both languages. Therefore, they are initially tagged as 'ambiguous'. The rank threshold at which tokens are classified as 'ambiguous' was iteratively determined to be 702 for EN-ES, 127 for EN-NL and 112 for DE-TR. This tag 'ambiguous' is only temporary, and once the classification process has been completed it is reassigned to the majority language found in the Tweet.

In the second additional step, a context-based rule is introduced to account for tokens that are being misclassified because they are orthographically identical and frequent in both languages, but are not categorized as grammatical function words. In these cases, if the language of both the preceding and following word is the same, the token is reassigned to match that language. This step accounts for words that are borrowed from another language and have been integrated into the lexicon and should therefore not be classified as codeswitching. However, this step is only incorporated if the ranks of the particular token in the respective dictionaries are sufficiently similar. The maximum distance between the ranks was iteratively determined for each language pair and is 16.000 for EN-ES, 27.000 for EN-NL and 0 for DE-TR.

5 Results

Table 1 shows the results of the classification process described above. Note that these figures include all tokens found in the Tweets that have been labeled as either one of the languages in the language pair. This does not include Twitter handles and hashtags or emoticons, as these were classified as 'other' and excluded from further evaluation. In general, the performance of the classifier has exceeded our prior expectations, with an F1 as high as 0.963 and 0.983 for the Spanish-English Tweets. However, it is also evident that there are still some challenges to overcome and that each language pair has particular characteristics that influence the performance of the classifier.

Table 1. Evaluation of token-based classification

	Spanish (EN-ES)	English (EN-ES)	English (EN-NL)	Dutch (EN-NL)	German (DE-TR)	Turkish (DE-TR)
P	.964	.985	.453	.995	.915	.939
R	.962	.980	.618	.822	.845	.771
F1	.963	.983	.524	.900	.879	.847

Spanish-English outperformed the other language pairs with precision and recall competitive to the state of the art as presented in Shirvani et al. (2016). Accounting for a significant proportion of misclassified tokens are either words containing irregular orthography, such as 'ooooooooommmmmmmg', 'noooooo',

and ‘meee’. All of these tokens could be normalised in a pre-processing step in order to improve performance. Having said that, the majority of words containing deviating spelling or abbreviations, for example ‘jajajaj’ (‘hahaha’) in Spanish and ‘btw’ (‘by the way’) in English, are actually captured by the classifier. This suggests that the incorporation of both the Wikipedia article and the comment section is an important element in this classification approach. Furthermore, some words were misclassified because they are present in both languages. Examples of homonyms include the verb forms ‘prove’, ‘embrace’ and ‘continue’. The status of several other tokens, such as ‘ugh’, ‘ahh’, ‘pfft’ and ‘wey’, could be contested as they do not strictly belong to either language. Interestingly, the noun ‘broncos’ was false-positively identified as English, probably based on the occurrences of the Denver American Football team in the EnglishWikipedia. The classifier does not have a separate named entities tagger and named entities are considered to be part of the language of origin. This means that named entities correctly identified as Spanish are for example ‘san antonio’, ‘gloria trevi’ and ‘san marcos’.

Turkish-German was the second best performing language pair. The Turkish dictionary was the smallest one available, but since Turkish and German are part of very different language families and are the least similar, the pair performed with F1 scores of 0.847 and 0.879. Similar to the Spanish-English corpus, misclassified tokens consist of words with incorrect or irregular spelling, such as ‘*verstandnis’ (‘understanding’), ‘*nasilsn’ (‘how are you’), ‘*anlatacann’ (abbr. ‘you will tell’) and ‘*seniiii gelisiniiii bekliyorum gözleee’ (‘looking forward to seeing from you soon’), while ‘*insallahhhh’ (‘god willing’) and ‘*anladiiim’ (‘i understand’) were correctly identified as Turkish. Named identities assigned to the correct language by the classifier include ‘Bahar’, ‘Sezen Aksu’, ‘Ezel’, ‘Tolga Cigerci’, ‘Frau Geiger’, ‘Galatasaray’ and ‘Bochum’. The small size of the Turkish dictionary, however, had obvious influence on recall for Turkish as several inflected forms of otherwise common lemmata such as ‘kıyımıza’ (‘to our shores’) and ‘domuzköpeği’ (ad-hoc literal translation from German idiom ‘innerer Schweinehund’, ‘inner temptation’) were not originally recognized as Turkish by the dictionary look-up, but reassigned to Turkish by the context rule. This rule did not manage to capture all such forms evading the dictionary based assignment and failed with tokens such as ‘*yalmazdım’ (‘i would not write’), ‘*varediyorsun’ (‘you create’), ‘çiğdemim’ (‘my crocus’, ‘my love’). These examples highlight a characteristic property of Turkish: as an agglutinating language, Turkish attaches a broad range of particles, for example the prepositions ‘de’/‘da’ (‘in’, ‘with’) or possessive pronouns like ‘im’ (my), directly to the words in the open word classes. However, these patterns are highly regular and thus accessible for resolution through parsing. Applying a morphological parser to words not found in the base lexicon might thus greatly improve recall on inflected Turkish words or phrases and compensate for the lack of entries in the smaller dictionary. Such a parser-based dictionary compensation mechanism might additionally improve recall on intra-word level codeswitched tokens such as ‘partyler’ (‘party’ + plural morpheme), ‘deutschlandda’ (‘germany’ +

preposition 'in') and 'schatzim' ('darling' + possessive pronoun 'my'), which are all German lemmata combined with Turkish morphemes denoting plural, location and possession. Çetinoğlu (2016) gives further examples of intra-word code-switching in the corpus.

Out of the three language pairs examined in this paper, English-Dutch is the most similar. It must also be taken into account that the English-Dutch corpus was constructed to examine code mixing more generally, rather than to evaluate automatic language classifiers. This means that there is less CS in this corpus than in the other two language pairs and there are more single word inclusions as opposed to intra-sentential CS. In general, Dutch contains many words that have been borrowed from English and have either been fully integrated into the Dutch vocabulary or are used even though there are Dutch equivalents. Consequently, the Dutch dictionary contains many English words. Words such as 'arrogant', 'stress', 'weekend', and 'incident', have been tagged as English, but they should be classified as Dutch because there are no other Dutch equivalents and therefore they cannot be considered to be CS. Contrastingly, words such as 'happy', 'same', and 'highlights', are English words with Dutch equivalents that are used frequently in the Dutch language, but are incorrectly classified as Dutch. They should be classified as English and considered to be CS, but are misclassified due to the context rule. The many overlapping lexical items explain why the F1 score for English is much lower when combined with Dutch than it is with Spanish.

6 Conclusion and Future Work

We presented a simple dictionary-based classification system for the identification of CS on Twitter. The results for Spanish-English are comparable to current state of the art classifiers even though the approach taken in this paper is much more simplistic. The classifier does not need any external toolkits or additional features such as a POS tagger or suffix information as it relies solely on the frequency information of the tokens within each dictionary. The use of Wikipedia as a dictionary resource has allowed for the classification of formal language as well as the colloquial language that is characteristic of language on social media. The classifier managed to successfully identify the language of sequences such as 'jajajaj' and 'omg' automatically. Nevertheless, many irregular tokens were not identified correctly and the addition of a token simplification rule, to reduce sequences such as 'noooooo' to 'no', would improve performance. This approach can be adapted to any language as long as the resources on Wikipedia are available and of an appropriate size. The difficulty lies in finding appropriate CS material on which to train and test the classifier. Once new Twitter corpora containing CS have been created, we plan on incorporating a wider variety of languages and focusing on how to improve the classification of closely related languages.

References

Bullock, B.E., Toribio, A.J.: Themes in the study of code-switching. In: Cambridge Handbooks in Language and Linguistics, pp. 1–18. Cambridge University Press (2009). https://doi.org/10.1017/CBO9780511576331.002

Çetinoglu, Ö.: A Turkish-German code-switching corpus. In: LREC (2016)

Chanda, A., Das, D., Mazumdar, C.: Unraveling the English-Bengali code-mixing phenomenon. In: EMNLP 2016, p. 80 (2016)

Dongen, N.: Analysis and prediction of Dutch-English code-switching in Dutch social media messages. Master's thesis, Universiteit van Amsterdam (2017)

Jaech, A., Mulcaire, G., Hathi, S., Ostendorf, M., Smith, N.A.: A neural model for language identification in code-switched tweets. In: EMNLP 2016, p. 60 (2016)

Maharjan, S., Blair, E., Bethard, S., Solorio, T.: Developing language-tagged corpora for code-switching tweets. In: LAW@ NAACL-HLT, pp. 72–84 (2015)

Molina, G., AlGhamdi, F., Ghoneim, M., Hawwari, A., Rey-Villamizar, N., Diab, M., Solorio, T.: Overview for the second shared task on language identification in code-switched data. In: Proceedings of the Second Workshop on Computational Approaches to Code Switching, pp. 40–49 (2016)

Muysken, P.: Bilingual Speech: A Typology of Code-Mixing, vol. 11. Cambridge University Press, Cambridge (2000)

Shirvani, R., Piergallini, M., Gautam, G.S., Chouikha, M.: The Howard University system submission for the shared task in language identification in Spanish-English codeswitching. In: Proceedings of The Second Workshop on Computational Approaches to Code Switching, pp. 116–120 (2016)

Solorio, T., Blair, E., Maharjan, S., Bethard, S., Diab, M., Ghoneim, M., Hawwari, A., AlGhamdi, F., Hirschberg, J., Chang, A., et al.: Overview for the first shared task on language identification in code-switched data. In: Proceedings of the First Workshop on Computational Approaches to Code Switching, pp. 62–72 (2014)