Language Identification of Words in Code-Switching Text with Recurrent Neural Networks

Jorge Hernandez
University of Texas at Austin

Daniel Loera
University of Texas at Austin

Abstract

Code-switching is a naturally occurring phenomenon where bilinguals or multilinguals switch between their languages while speaking or writing. In this project, we build a Recurrent Neural Network model that can identify the language of every word in an text which exhibits code-switching. Specifically, the model will be labeling words as belonging to either English, Spanish, or other. As baseline models, we utilize a frequency based classifier and a Naive Bayes classifier. In addition, we also utilize three different datasets and compare each of our model's performance across various types of text. Overall, the RNN model outperformed all other models across the datasets. Furthermore, the RNN model is able to utilize surrounding context to more reliably classify interlingual homographs. However, there is still some work to be done to label homographs more consistently.

1 Introduction

This project aims to construct a model that can provide an accurate and stable foundation for identifying the language of words in English-Spanish code-switching text. Frequency-based models can produce great surface-level results, but they struggle when the text includes interlingual homographs. Homographs are similar to cognates in that the pair of words have similar form, but they have a different meanings across languages. That being said, cognates won't be of interest in the present project. This is precisely due to the similarity in both form and meaning, which makes the correct label assignment ambiguous. Take for instance the English-Spanish cognate 'me' and its usage in the sentence "No me digas" (translation: "Don't tell me"). Technically speaking, 'me' could be labeled as either English or Spanish at least when only considering the written text. It could be the case that when considering auditory information it is uncommon for the intended 'me' to be the English version and thus making the Spanish version the more correct version. However, this is outside the scope of the present project.

Moving on, frequency-based models cannot overcome the homograph issue in a practical way since identifying the language of homographs requires analyzing the surrounding words' languages. For example, the text "I rolled the dice" contains the homograph "dice". The Spanish meaning of "dice" roughly means "says", so it is used in entirely different contexts than its English counterpart. This vast difference in context can be used to determine if code-switching was taking place or a homograph suddenly appeared. This is precisely why this project uses a recurrent neural network (RNN) as the foundation to predict these labels. With the use of short-term memory in the forward and backward directions, RNNs provide an excellent data structure to possibly give more accurate predictions in code-switching text.

2 Design & Procedure

2.1 Data

There are two main datasets used in this project. One dataset is provided by EMNLP (2014) and originates from Twitter. The other dataset is provided by BangorTalk from Bangor University (2008) and originates from a series of chat logs of recorded conversation. A third dataset was generated by combining the previously mentioned datasets. We label these datasets as *twitter*, *chat*, and *combined*.

Since both of these datasets came from two entirely different sources, they had two sets of labels for words. We consolidated and merged all labels into the following set:

en: English words

es: Spanish words

other: named entities, symbols, mixed, or ambiguous words

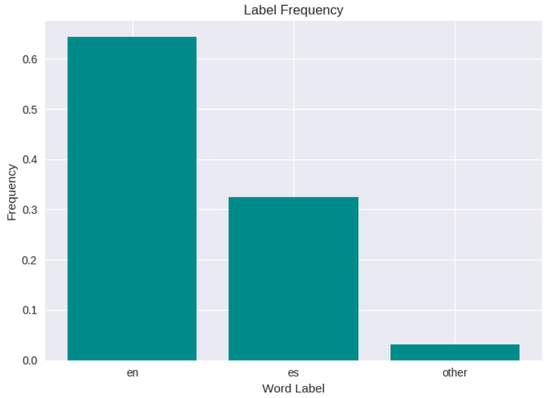


Figure 1: Twitter data label frequency

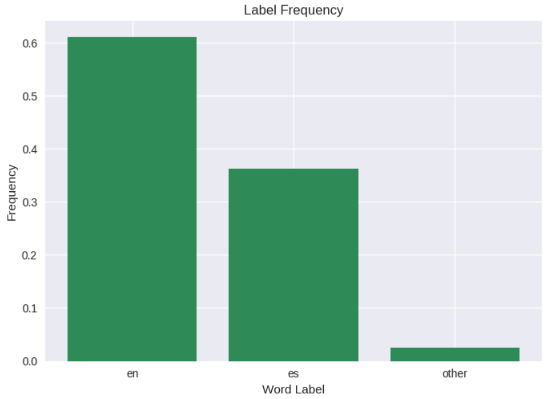


Figure 2: Chat data label frequency

The twitter and chat datasets showed significant signs of class imbalance as shown in Figures 1 and 2. There seems to be almost double the amount of English occurrences than Spanish occurrences. This in turn also means that the combined dataset also experiences some class imbalance.

2.2 Models

Frequency

This model predicts labels by simply referring to an internal count of previously seen words and their associated labels. The prediction will be whichever label is most frequently associated with a particular word. This is possibly the most naive approach as it depends on words being language-exclusive, which is why it struggles with homographs.

Naive Bayes

This model predicts labels by using Bayes' theorem on a character level to predict based on its morphological structure.

i.e. given that W= set of characters to label

$$\prod_{i=1}^{n} P(\text{character} = W_i | \text{ label} = l)$$

This model is slightly more sophisticated than Frequency as it can identify words based on character sequences that might be unique to one language. Again, this model cannot take into account surrounding context to determine a word's language.

RNN

The RNN model was built with context in mind. That is, as the network progresses through a sequence of input it retains some bit of information from its previous timestep. This makes it ideal to work with code-switching text as, most of the time, context is needed to accurately label words.

The RNN used is this project uses an LSTM and learns its own embeddings where each embedding vector is length 200. Also, this RNN contains two LSTM layers that are bidirectional, meaning it scans the text from the left-to-right and right-to-left directions. Below is the full layer structure of the RNN.

$$\text{Embedding} \rightarrow \text{LSTM} \rightarrow \text{Dropout} \rightarrow \text{LSTM} \rightarrow \text{Decoder (Linear Layer)}.$$

2.3 Training & Testing

In addition to overall language identification accuracy (English, Spanish, or other), our project specifically focused on improving the precision, recall, and F1 for English and Spanish identification. When testing the RNN, if it must label a word that is not present in its vocabulary, we use the embedding for the word with the smallest edit distance.

After all models were tested and evaluated, we chose the highest performing one and shifted the weight of the **es** label to counteract the class imbalance present in both datasets.

3 Results

Due to the large amounts of results collected this section provides a series of tables that summarize the performance of a model under different training and testing conditions. Before defining the metrics used in the present project we'll first discuss a bit of terminology. Take X to be either the label **en** (English), **es** (Spanish), or **other**. **True positives** are observations which were correctly labeled as X. **False positives** are observations, which were labeled as X, but are in fact some other label. **False negatives** are observations, which were labeled something other than X, but were indeed X. Moving on, precision is defined to be the fraction true positives over the sum of true positives and false positives. Recall is defined to be the fraction of true positives over the sum of true positives and false negatives. The overall accuracy is simply the number of observations that were labeled correctly over the total number of observations. For the purposes of this project only metrics for English and Spanish are shown below. All metrics are to be interpreted as percentages.

3.1 Frequency Model

Dataset (lang)	Precision	Recall	F1
Twitter (en)	96.202	89.537	92.750
Twitter (es)	95.154	76.289	84.683
Chat (en)	92.546	95.396	93.949
Chat (es)	95.558	83.560	89.157
Combined (en)	93.665	94.170	93.917
Combined (es)	95.349	83.729	89.162

Table 1: Performance for frequency model when trained and tested on various datasets.

Dataset	Overall Accuracy
Twitter	85.038
Chat	90.815
Combined	90.080

Table 2: Accuracy performance when identifying all labels (en, es, other).

3.2 Naive Bayes

Dataset (lang)	Precision	Recall	F1
Twitter (en)	88.486	98.250	93.113
Twitter (es)	90.981	78.582	84.328
Chat (en)	87.028	97.152	91.812
Chat (es)	92.289	78.593	84.892
Combined (en)	87.497	97.398	92.182
Combined (es)	92.079	79.900	85.558

Table 3: Performance for Naive Bayes model when trained and tested on various datasets.

Dataset	Overall Accuracy
Twitter	89.154
Chat	88.609
Combined	88.913

Table 4: Accuracy performance when identifying all labels (en, es, other).

3.3 RNN

Dataset (lang)	Precision	Recall	F1
Twitter (en)	94.367	96.474	95.409
Twitter (es)	91.207	91.414	91.310
Chat (en)	95.290	95.763	95.526
Chat (es)	92.788	93.035	92.911
Combined (en)	93.931	97.915	95.881
Combined (es)	96.400	90.493	93.353

Table 5: Performance for Naive Bayes model when trained and tested on various datasets.

Dataset	Overall Accuracy
Twitter	92.630
Chat	93.933
Combined	94.141

Table 6: Accuracy performance when identifying all labels (en, es, other).

3.4 Weighted RNN

We chose the RNN model trained on the combined dataset as the best model based on overall accuracy and F1 score. Then the label weighting was changed to handle the class imbalance in the data. Since Spanish words were in the minority across datasets, it was weighted 20% higher than English. The **other** label also experienced class imbalance, but the focus of this project is to correctly predict Spanish and English words. Hence, it was weighted the same as English. The overall accuracy for this version of the model was 93.491%. Below are the results for the weighted version of the model.

Language	Precision	Recall	F1
English	95.286	95.315	95.300
Spanish	92.081	93.199	92.636

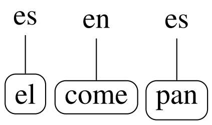
Table 7: Performance for weighted RNN model when trained and tested on the combined dataset.

4 Discussion

4.1 Frequency Analysis

The frequency model served as a baseline for this project and performed quite well. The best two frequency models were trained on the combined and chat datasets with one being only slightly better than the other in some metrics (see Table 1). It was at significantly better than randomly guessing the label for each observation. This is to be expected, since the strategy this model employs is similar to having two complete dictionaries for English and Spanish. Of course the difference here lies in that the generated dictionaries are relative to the data. This means that the frequency model would have trouble predicting words that were not present in the data it was trained on. This drawback could be mitigated by training on a dataset that uses a larger English and Spanish vocabulary; however, one problem still remains.

Overall, this model proves quite effective for words that are exclusive to one language. Unfortunately, English and Spanish have interlingual homographs. Following the strategy of the frequency model, a homograph will only be labeled correctly if the the correct language label happens to be the most frequent in the dataset. As an example, the text "El come pan" was ran through the frequency model that was trained on the chat data.



As shown above *come* is labeled as English when it should be Spanish. The Spanish meaning of *come* translates to "eat", so the entire sentence translates to "He eats bread". This is due to *come* appearing more often as an English word rather than a Spanish word. Also, the version of the homograph being used is dependent on context, which is something this model does not make use of.

This model is also heavily affected by the class imbalance that is present in the dataset. We can see that from Table 1 the recall and F1 score for Spanish words across datasets is lower than the English counterparts. The same can't necessarily be said for precision.

4.2 Naive Bayes Analysis

The Naive Bayes model performed the best when trained on the chat dataset. This model struggled to maintain an accuracy over 90%, and was the worst model overall but is still a great model when compared to randomly guessing. This model is much less promising probably due to the fact that it relies solely on the morphology of words (due to analyzing at the character level), rather than syntactical information from the text. While there might be certain sequences that are exclusive to a language, (e.g. the sequence "kn" is exclusive to English) English and Spanish share many word beginnings, endings, and other structures in between.

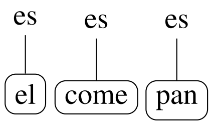
Overall, this methodology is perhaps too ambitious. English and Spanish share many linguistic roots, and assuming there might be a universal morphological identifier in each word is a bit too optimistic. Additionally, this model has no useful features that will assist in the identification of homographs. It is essentially "blind" to the entire context of the text–where some of the most important information lies.

4.3 RNN Analysis

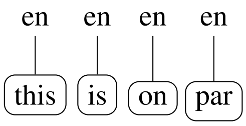
The RNN model was by far the best model used in the present project. All of the metrics shown in tables 5 and 6 are above 90%. When comparing the RNN model and the frequency model we see that the RNN model has better recall for Spanish observations under every dataset. Therefore, the frequency model is affected by the class imbalance problem more than the RNN model. This is probably due to the RNN's inherit usage of surrounding context. It also helps that the RNN is scanning the text in the forwards and backwards direction. However, there is perhaps still some effect of the class imbalance problem due to the Spanish metrics falling a bit behind the English ones as shown in Table 5. To try to resolve this we trained the best RNN model with different label weightings.

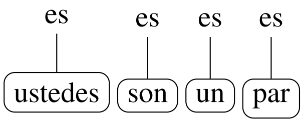
When comparing tables 5 and 7 we see that only the Spanish recall and English precision improved. The reduction in Spanish precision and English recall shows that this trade off did not necessarily solve the class imbalance problem in the most efficient way possible.

This model shows strides in improvement towards labeling text that includes homographs. Specifically, this model does great in labeling homographs when the sentence is monolingual, but struggles when code-switching is added into the mix. Examples of this can be seen in the diagrams below.

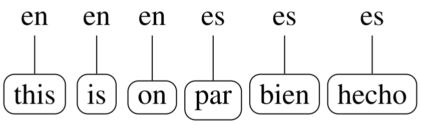


As shown above, the RNN correctly labels all words as Spanish, unlike previously seen in the frequency model. Another promising example of homograph monolingual predictions can be seen with the examples below.





Both examples above show correctly labeled monolingual sentences thanks to the use of prior context to identify the text's primary language. "par" in the English sentence is referring to the golfing terminology while the Spanish word roughly translates to "pair". The entire Spanish expression translates to "You (pl.) are a pair". Trying to label "par" in the presence of code-switching proves to be a challenge for the RNN model. An example of the code-switching case can be seen below.



The text above roughly translates to "This is on par, great job". The model fails to recognize that *par* is actually is in English. There does not seem to be a reliable way to make sense of this this error, but it seems as if the RNN is choosing the label for the homograph that it is **most likely** to be associated with, given the context. It cannot be said for sure, as we need more labeled homograph code-switching data, as there are numerous examples to the contrary.

We see that the RNN model is a step in the right direction for this task, but there are some areas that still need improvement, particularly when it comes to labeling homographs. Perhaps, different features are needed to handle this case. The usage of part of speech tags would be helpful, presumably, since homographs could be part of different word classes across languages.

5 Conclusion

In this paper we have created an RNN, which is able to classify words as belonging to either English or Spanish by training code-switching text with better performance than naive solutions. This model has a limitation when labeling interlingual homographs. Moving forward, different types of features should be employed that include more linguistic context. Specifically the usage of part of speech tags would handle cases where the homograph has different parts of speech across languages. In addition, we propose the generation of a more representative dataset that includes a more diverse English and Spanish vocabulary as well as a vast amount of code-switching text that includes homographs.

References

BangorTalk. 2008. Corpus: Miami. http://bangortalk.org.uk/speakers.php?c=miami. Accessed on 2019-05-01 from Bangor University.

Conference on Empirical Methods in Natural Language Processing. 2014. First workshop on computational approaches to code switching. http://emnlp2014.org/workshops/CodeSwitch/call.html. Accessed on 2019-05-01.