Loanword Detection with Maximally Simple Tools

Michael Hammond

Dept. of Linguistics, U. of Arizona, Tucson, AZ 85721, USA

Abstract

This paper demonstrates a maximally simple approach to loanword detection. The different systems developed here all start with handcrafted features. The first approach used a logistic regression model and a second approach used a simple feed-forward neural network. Finally, in a third approach, the output of the neural network is "repaired" in various ways for the submission. The resulting system is not competitive, but exemplifies how extremely simple techniques can produce output that is "in the ballpark".

Keywords

loanwords, English, Spanish, ensemble methods

1. Overview

Loanword detection involves identifying loanwords from one language in another. Here is a sample where several borrowings from English (boldfaced below) appear in a Spanish sentence.

> No me llevó mucho tiempo encontrar un grupo de **hashtags** y **posts** nostálgicos de la guerra civil.

In the simple case, we might proceed with wordlists for both languages: L_s and L_e. If a word appears in L_s, but not L_e, it is Spanish; if it appears in L_e and not L_s, it is English.

Solutions based on wordlists are notoriously brittle, however. The set of words for any language is huge, if not infinite. Therefore the absence of a word in, say, L_s, may be because the word isn't Spanish, but it may instead be because the word simply hasn't been recorded (yet) as a word of Spanish.

Another problem is that some items may appear in *both* lists. For example, the string *local* may be an instance of the English adjective (pronounced [lókəl]) or it may be an instance of the Spanish adjective (pronounced [lokál]).

There are other thorny problems as well. For example, if the English language author Shakespeare is mentioned in a sentence in Spanish, is that a borrowing or simply a name?

Especially challenging are loanwords that have undergone *adaptation* either in spelling or orthography. For example, Spanish *beisbol* for English *baseball* where the spelling has been changed or Spanish *líderes* for English *leaders* where the spelling has changed and the form exhibits the appropriate Spanish plural form.

This is thus a hard task.

2. Previous approaches

A very similar task was was run in 2021 [1]. In that task, teams were provided specific training data. At that time, only two system descriptions were included. One team used conditional random fields and data augmentation [2]. The other team used used pretraining with unlabeled data [3, 4].

The task is related to the more general task of language identification [5, 6, 7]. There are a few earlier studies in in different languages using a variety of more traditional techniques [8, 9, 10].

IberLEF 2025, September 2025, Zaragoza, Spain

hammond@arizona.edu (M. Hammond)
https://hammondm.github.io (M. Hammond)
0000-0001-5072-7150 (M. Hammond)

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073
published 2025-12-10

3. The specific task

The specific task here was to identify unassimilated borrowed words or word sequences from English in Spanish sentences [11, 12]. We were given a model file `reference.csv` demonstrating the format. Here is the first line of that file.[1]

```
Somos un país en el que ‘youtubers’ y ‘gamers’ millonarios deciden
irse al paraíso fiscal de Andorra porque tributan menos sin preocuparse
del bienestar de sus vecinos y de quienes les han hecho ricos,
ni acordarse de la Educación, Sanidad, infraestructuras de las
que han disfrutado durante años gracias a la solidaridad de todos.
;youtubers;gamers;;;
```

Fields are separated with `;`. The sentence appears first and borrowings occupy remaining fields on the line. In this line, the borrowings appear in single quotes, but this is not generally true.

Testing was on the basis of the `input.csv` file (from the submission website). This file also has a single sentence on each line, but the remaining fields that would otherwise contain any borrowings are absent.

The goal was to extract the borrowings from the sentence and append them as additional fields on the line.

4. The approach

The approach taken here is maximally simple. I deliberately make use of the simplest possible techniques with an eye to seeing how far these go. There is a pedagogical purpose to this. I teach an introductory course in our *Human Language Technology* program on statistical natural language processing. The question addressed here is whether techniques covered in that introductory course are sufficient for a reasonable attempt at loanword identification.

I do not expect these to be competitive; rather the hope is that: i) they are in the ballpark as far as success; and ii) that their performance points the way toward more successful strategies.

The specific strategy employed here is as follows.

1. Identify *by hand* features that are likely to be helpful in identifying borrowings.
2. Break each sentence up into words and do predictions word by word.
3. Build an initial logistic regression model to identify borrowings.
4. Build a simple neural net, replacing the logistic regression model.
5. Look at the output and add post-processing as needed.
6. Reasssemble words into sentences for evaluation.

5. Resources

My submissions all made use of these resources:

1. COALAS dataset (https://github.com/lirondos/coalas).
2. NLTK wordlists
   - `nltk.corpus.words` (English)
   - `nltk.corpus.cess_esp` (Spanish)
3. NLTK stemmers

[1] Here is a description of the task https://adobo-task.github.io/borrowing.html and here is the submission website https://www.codabench.org/competitions/7284/

    - `nltk.stem.PorterStemmer` (for English)
    - `nltk.stem.SnowballStemmer` (for Spanish)

4. unix/mac English wordlist (`/usr/share/dict/words`)
5. Another English wordlist (https://github.com/dwyl/english-words)
6. `spacy` taggers for English and Spanish

The COALAS dataset was used to train all models and is organized by word. Here's the first few lines of the training partition.

```
Microsoft   O
promete     O
formación   O
digital     O
gratis      O
a           O
25          O
millones    O
de          O
personas    O
este        O
año         O

El          O
gigante     O
del         O
software    B-ENG
Microsoft   O
lanzó       O
este        O
martes      O
```

Each word appears on its own line and is coded for whether it's a borrowing or not. Sentences are separated by spaces.

6. Features

I set this up initially as binary logistic regression where 0 is Spanish and 1 is borrowed. I generated a set of features based entirely on intuition. These include:

- *Is the word in all capitals?*
  The logic here was that capitalization would indicate an acronym.
- *Is the first letter capitalized?*
  This assumes proper names are not borrowings.
- *Does the word include non-alphabetic characters?*
  Strings containing special characters are not borrowings.
- *Does the word include common Spanish suffixes?*
  Check specifically for whether the word ends in high-frequency Spanish endings, e.g. *ión*, *o(s)*, *a(s)*.
- *Does the word appear in any of the English wordlists?*
  Brute-force check against all the English wordlists, unioned together and implemented as a python `set`.

- *Does the word appear in the Spanish wordlist?*
  Same for Spanish.
- *What is the character bigram probability with respect to English?*
  I built a character bigram model for both languages and this feature was the score for the English model.
- *What is the character bigram probability with respect to Spanish?*
  Score for the Spanish bigram model.
- *Can the word be stemmed for English?*
  Using the `nltk` Porter stemmer, is the result "smaller" than the input?
- *Can the word be stemmed for Spanish?*
  Same procedure using the Snowball stemmer for Spanish.
- *I tagged the sentence for Spanish with 'spacy' and generated a rule for each tag.*
  I used the `spacy` part-of-speech tagger to find the tags for each word. I then had a rule for each tag. The basic idea here is that borrowings would be most associated with nouns and adjectives and these rules would pick up on that.

Both logistic regression and neural net models made use of these same features. With those rules in hand, each item in the COALAS dataset was used for training. Each item was scored with respect to the rules above and then converted to z-scores. Using the `reference.csv` file for testing, this got an F_1 around .6 over multiple runs. Results from these models were not submitted.

7. Simple neural net

The logistic regression model did not include interactions between factors, so I implemented a neural net with `pytorch`. The logic here is that a fully-connected multi-layer net would capture all possible interactions between features. The architecture was extremely simple: 3 layers with the same dimensions as the input. There was a final output layer that produced a single output value. The activation function for all layers was sigmoid.

Again I tested with `reference.csv` and various numbers of epochs and different batch sizes, this reached F_1 values as high as .65 over multiple runs.

8. Rules

I augmented the neural net with a rule-based system. That is, I applied rules to the output of the network. Specifically:

- *Adjacent borrowings count as a single borrowing.*
  Since my system was word-based, I needed to convert the output to back to sentences and concatenate borrowings.
- *All quoted sequences of up to 3 words are borrowings.*
  It became clear that quotes were being used to mark borrowings in the `input.csv` file.
- *All capitalized sequences of up to 3 words are borrowings.*
  Capitalization was used in a similar fashion.
- *If a word appears in any of the English wordlists and does not appear in the Spanish wordlist, then it's a borrowing.*
  This required some tweaking. Specifically, if a word appeared in both English and Spanish lists *and* was adjacent to something already marked as a borrowing, then it was marked as a borrowing as well.

This approach reached an F_1 of about .75 over multiple runs.

9. Discussion

Let's look a bit more closely at the performance of the system. We focus on the final 15th run that I submitted. This achieved a precision score of 0.67, recall of 0.84 and F_1 of 0.75. In terms of exact numbers, there were 1735 true positives, 844 false positives, and 341 false negatives.

If we look over the actual errors on a sentence-by-sentence basis, we find 680 sentences out of 1836 had some sort of error. Of these, the majority were cases where there were additional borrowings to be found and our system did not find them.

Another pattern was observed as well. These were a number of cases where a borrowed span was redundantly parsed or separately parsed. Here's an example (sentence, desired result, actual result):

```
Falsificó los papeles y la policía acabó deteniéndole en el control de aduana,
cuando intentó cruzar la frontera con una GREEN CARD FAKE.
['GREEN CARD', 'FAKE']
['GREEN', 'CARD', 'GREEN CARD FAKE', 'FAKE']
```

A number of these could be mitigated by checking the output for repeated items, though we'd have to investigate what to do when this occurs, which items to remove when repetitions occur.

10. Conclusion

In the approach here, there were three principal components. First, features were generated by hand. Second, I used a simple neural network to model how the features might interact. Third, I included additional rules to postprocess the output of the network.

Again, the goal of this system was to exemplify simple techniques taught in my introductory statistical natural language processing course.

There are a number of ways we might improve on this system. First, creating the features by hand is a bottleneck on how the shape and context of each word might determine whether it's a borrowing. A better approach would be to input the full spelling of the word and the words in context and use that information to generate features.

A second and related move would be to enrich the neural net architecture, specifically something that included the context of each word would surely help. A recurrent or attention-based architecture are the obvious choices here.

Acknowledgments

Thanks to Diane Ohala for useful discussion. All errors are my own.

Declaration on Generative AI

The author has not employed any Generative AI tools.

References

[1] E. Á. Mellado, L. E. Anke, J. G. Arroyo, C. Lignos, J. P. Zamorano, Overview of ADoBo 2021: Automatic detection of unassimilated borrowings in the Spanish press, arXiv preprint arXiv:2110.15682 (2021).

[2] S. Jiang, T. Cui, Y. Fu, N. Lin, J. Xiang, BERT4EVER at ADoBo 2021: Detection of borrowings in the Spanish language using pseudo-label technology, in: IberLEF@ SEPLN, 2021, pp. 940–946.

[3] J. De la Rosa, The futility of STILTs for the classification of lexical borrowings in Spanish, arXiv preprint arXiv:2109.08607 (2021).

[4] J. Phang, T. Févry, S. R. Bowman, Sentence encoders on stilts: Supplementary training on intermediate labeled-data tasks, arXiv preprint arXiv:1811.01088 (2018).

[5] K. R. Beesley, Language identifier: A computer program for automatic natural-language identification of on-line text, in: Proceedings of the 29th annual conference of the American Translators Association, volume 47, 1988.

[6] W. B. Cavnar, J. M. Trenkle, N-gram-based text categorization, in: Proceedings of SDAIR-94, Third Annual Symposium on Document Analysis and Information Retrieval, 1994, pp. 161–175.

[7] T. Dunning, Statistical identification of language, Technical Report, New Mexico State University, 1994. CRL MCCS-94-273.

[8] B. Alex, Automatic detection of English inclusions in mixed-lingual data with an application to parsing, Ph.D. thesis, University of Edinburgh, 2008.

[9] C. Furiassi, K. Hofland, The retrieval of false anglicisms in newspaper texts, in: Corpus linguistics 25 years on, Brill, 2007, pp. 347–363.

[10] G. Andersen, Semi-automatic approaches to Anglicism detection in Norwegian corpus data, in: The anglicization of European lexis, John Benjamins Publishing Company, 2012, pp. 111–130.

[11] E. Álvarez-Mellado, J. Porta-Zamorano, C. Lignos, J. Gonzalo, Overview of ADoBo at IberLEF 2025: Automatic Detection of Anglicisms in Spanish, Procesamiento del Lenguaje Natural 75 (2025).

[12] J. Á. González-Barba, L. Chiruzzo, S. M. Jiménez-Zafra, Overview of IberLEF 2025: Natural Language Processing Challenges for Spanish and other Iberian Languages, in: Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2025), co-located with the 41st Conference of the Spanish Society for Natural Language Processing (SEPLN 2025), CEUR-WS. org, 2025.

A. Online Resources

- https://adobo-task.github.io/borrowing.html
- https://github.com/hammondm/adobo2025/tree/main
- https://faculty.sbs.arizona.edu/hammond/ling439539-sp25/