Dangerous Liaisons: the Impact of Phonetics in False Friends Detection Among Romance Languages

Francisco Martínez Lasaca
IT University of Copenhagen
frml@itu.dk

Abstract

False friends are words in different languages that are written or pronounced similarly, but differ significantly in their meaning. Their impact on second language learning is significant, but generating lists of false friends is a difficult task that requires the manual work of experts. In this paper we analyze the impact of adding phonetic transcriptions to Machine Learning false friends models trained on Romance languages.

Keywords: false friends, Romance languages, fastText

1 Introduction

Background While learning a new language, the elements that are similar to the mother tongue will be simpler to learn than those that are different (Lado, 1957). The student will benefit from knowledge in their first language, which will apply to the new language being learned (Biennale and LeBlanc, 1989; Gass, 1988; Nation, 2003). A clear example are *cognates*, or words sharing a common origin, regardless of whether their meanings have evolved apart or not. In the case that both meanings match, words that are written similarly and have the same or very close meanings can be recognized (e.g. *verd* in Catalan, *vert* in French and *verde* in Spanish, Italian or Romanian). False friends (also known as *false cognates*) are those words which share or have close signifiers but differ as regards their meanings (Chamizo-Domínguez, 2012). An example: *pijnlijk* [ˈpɛinˌlək] means *painful* in Dutch, but *peinlich* [ˈpaɪnˌlɪç] means *embarrassing* in German.

Otwinowska and Szewczyk (2019) have given empirical evidence to the problem of false friends: they were the most difficult words to translate from a language being learned, if compared with cognates and non-cognates.[1] However, creating lists of false friends is hard, as they require the manual work of linguistics experts. Frunza and Inkpen (2009) created a Machine Learning method to identify cognates and false friends between French and English using already existing gold standard data, but they did not took into account the phonetic transcriptions of words.

Objective In this project, we explore how the phonetic similarity of the Romance languages (Boyd-Bowman, 1980) can benefit false friends detection.

2 Data

The data are corpora of aligned word embeddings in different languages. The source is fastText,[2] a repository that contains corpora for more that 120 languages. The word embeddings are first obtained from Wikipedia[3] dumps in different languages, which are processed in a Skip-gram model (Bojanowski et al., 2017). Then the word embeddings for the different languages are aligned as shown in (Joulin et al., 2018). Each language has a dataset of 1 million words, with vector embeddings of 300 components.

fastText's data is ideal for this project, as their aligned embeddings overtake in many NLP tasks and words that are semantically similar have close embeddings.

3 Features

We want to detect false friends among pairs of words in different languages. This relies on comparing words. And every word carries different pieces of information which serve as criteria to decide how close the words are to each other. We distinguish three of them:

[1] Non-cognates are words that have significantly different signifiers and different meanings in both of the languages

[2] https://fasttext.cc
[3] https://www.wikipedia.org

Spelling How a word is written. It comprises the sequence of letter it contains and the order in which they are placed.

Embedding What a word means. Even if their spellings are different, both the words *lorry* and *truck* share the same semantic value. We represent the semantic value of a word with its word embedding, which is a dense vector of float numbers.

Phonology How does a word sound. How we pronounce words is studied by the *phonology* of languages, and can be represented by their *phonetic transcriptions*, which is standardized by the International Phonetic Alphabet (IPA) (1999).

We will use the different features of the previous word characteristics to build our models.

4 Baseline

The baseline model only takes into consideration spelling and embedding. Two words are false friends if their spellings are similar but their embeddings are far apart.

5 Preprocessing

We perform two preprocessing steps: reducing the vocabulary size and the embedding dimensions.

Vocabulary size Each language dataset has 1 million words. If we were to compute the cartesian product of all the words in each language, that would generate 10^{12} instances, which is unfeasible. Instead, we extract 10 000 words from each language, reducing the number of pairs to 100 millions, which is still too much. Luckily, false friends are expected to have similar spellings. Thus, we only take the cartesian product of words which start by the same letter, reducing from 100 millions of pairs to 6 millions. This huge reduction will allow us to experiment even if we are not taking into account all the theoretically possible pairs.

Embedding dimensions The word embeddings of fastText have 300 components. Having such long embedding representations allow representing very subtle characteristics of words, but it has the drawback that it can slow down computations. To solve this, we can use a dimensionality reduction algorithm such as the Principal Component Analysis, or PCA (1901). A plot of the explained variance ratio while increasing the number of components can be found in Figure 1. 95 % of the variance can be explained with 176 components. However, we need to be cautious. We do not want to jeopardize the embedding distances. Fortunately, we observe that there is no impact on reducing the number of components. In fact, in Figure 2, we see that we can even reduce the number of word embedding components to 50 without breaking the word embeddings.

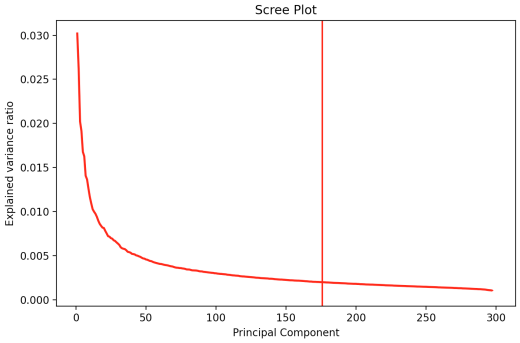

Figure 1: PCA scree plot

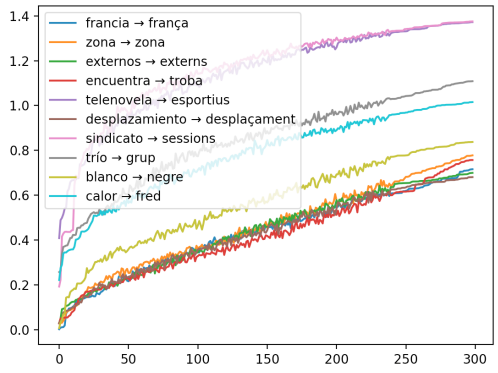

Figure 2: Edit distance over number of word embedding components. The pairs (*francia*, *França*), (*zona*, *zona*), (*externos*, *externs*), (*encuentra*, *troba*) and (*desplazamiento*, *desplaçament*) are synonyms in Spanish and Catalan. Note how all of them show the lowest embeddings edit distance. The pairs (*trío*, *grup*), (*blanco*, *negre*), (*calor*, *fred*) are words within the same semantic field (*trio* with *group*, *black* with *white* and *hot* with *cold*). The remaining pairs have no semantic similarity.

6 Features implementation

Once that we have reduced the vocabulary and the number of dimensions of the embeddings, we can calculate features on the pairs of words that will allow us to distinguish false friends. Recall that

they fall into three categories: spelling, embedding and phonology.

Measuring spelling distance Words, as strings, can benefit of well known edit distance algorithms, such as Hamming distance, longest common subsequence or the Levenshtein distance (1966), which we are going to use. In particular, we use normalized Levenshtein similarity, which returns a value in the $[0, 1]$ range meaning that the words are lexically far apart for 0, and identical for 1.

Measuring embedding distance We use two methods to determine how close are two words: Euclidean distance and cosine similarity. Given two word embeddings u and v with n components, their Euclidean distance is given by

$$\text{EUC}(u, v) = \sqrt{\sum_{i=1}^{n}(u_i - v_i)^2},$$

and their cosine similarity by

$$\cos(u, v) = \frac{\sum_{i=1}^{n} u_i v_i}{\sqrt{\sum_{i=1}^{n} u_i^2}\sqrt{\sum_{i=1}^{n} v_i^2}}.$$

Note that Euclidean distance *is* a metric and the cosine similarity is not, but it does not really matter for our purpose. In Figure 3 we present a histogram that plots many pairs of words in terms of their Euclidean distance and their cosine similarity. Both the variables have a coefficient of determination of $R^2 = 0.9949$, which means that either of the measures fits really well the other. Thus, we can use any Euclidean distance or cosine similarity indistinctly in this context.

Measuring phonetic distance Before computing the distance among phonetic representations, first we have to obtain them. In this project we are going to use Epitran,[4] a state-of-the-art Grapheme-to-Phoneme (G2P) conversion model that allows transforming words in concrete languages to IPA transcriptions.

Regarding their distance, we are going to use regular Levenshtein distance and a special weighted edit distance. Romance languages have a small subset of vowel sounds, if compared for example with Germanic languages (1980). Despite having small phonetic differences as shown in Figure 4,

[4] https://pypi.org/project/epitran/

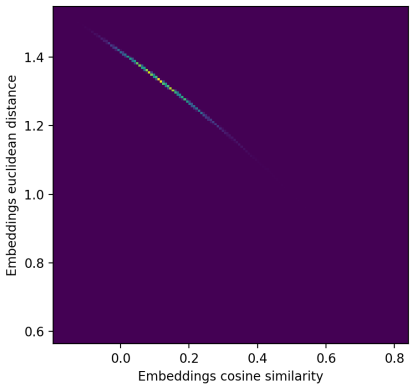

Figure 3: 2D histogram of word pairs

sounds with the same IPA transcription are recognized as allophones (that is, different phones that are perceived as the same sound by the speakers of a language). This allows us using IPA transcriptions as the starting point to calculate phonetic edit distance. However, assigning the same cost of replacing a vowel by a consonant than changing a vowel to another is not useful, as it ignores the way utterances are produced in the mouth (Wieling et al., 2009). Thus, the justification of using weighted phonetic edit distance. Our costs of substituting one letter per another different letter are presented in Table 1.

Substitution	**Cost**
consonant → vowel	2
vowel → consonant	2
consonant → consonant	1
vowel → vowel (Figure 5)	0.5 per edge

Table 1: Cost per letter substitution

The histograms of both the measures are presented in Figure 7 and 6, once the Levenshtein distances are normalized. We see that weighted phonetic distance follows a normal distribution, while phonetic normalized similarity loses many information while mapping many words to 0. We believe that this is because similar phonemes are not weighted properly and, consequently, are computed as completely different phonemes.

7 Model

Our model is built on top of the baseline model. A part of taking into consideration spelling and embedding (i.e. normalized Levenshtein similar-

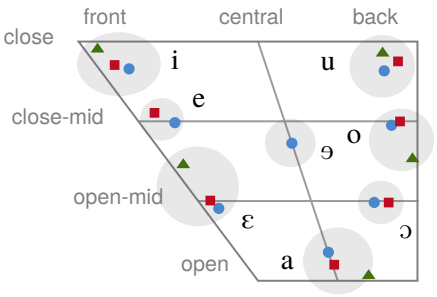

Figure 4: Vowel charts of Italian, Spanish and Catalan. Key: red/square = Italian (2004), green/triangle = Spanish (2010), blue/circle = Catalan (2009).

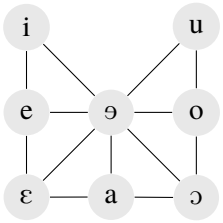

Figure 5: Vowel distances

ity and cosine similarity), it also uses weighted phonetic distances.

8 Analysis

The results of the baseline model are presented in Table 2: the more the normalized spelling edit similarity and cosine similarity thresholds are loosen, the higher chance to detect a pair of words as false friends. The results for the final model (which includes phonetic distance) are presented in Table 3. As we see, for many entries between both of the tables, the number of detected false friends is decreased.

We do not have lists of false friends for the

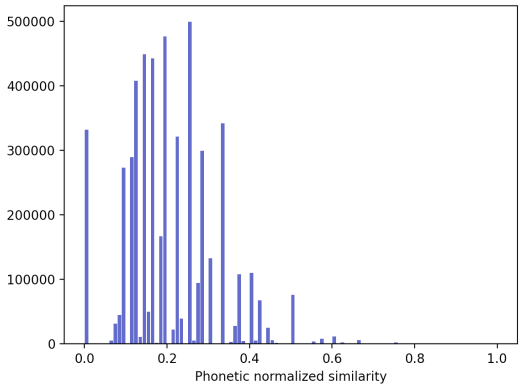

Figure 6: Normalized phonetic Levenshtein similarity histogram

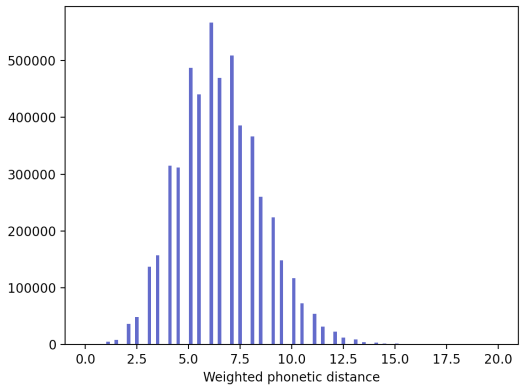

Figure 7: Weighted phonetic distance histogram

analyzed languages and, consequently, we cannot give measures comparing both results. However, a manual inspection of the words that are classified as false friends but that are not in the final model show an interesting fact. Many verbs are incorrectly tagged as false friends because their spellings are really close but their embeddings can be quite different because they are used in different situations (e.g. conjugations of *tú* and *usted*, that is, informal and formal *you*). Thanks to the phonetic criterion, we can filter them out (as shown in the descarted pair *incorporada* and *incorporar*, both parts of the verb *incorporate* in English).

However, there are still many false positives, such as verbal tenses or words that already exist in the target languages.

9 Conclusion

We have seen that phonetics help removing false friends while detecting false positives of false friends, but we have not measured its impact. We have seen that, even if built with simple rules, weighted phonetic distances are more informative than regular phonetic normalized similarities, as they take into account how words are pronounced more faithfully.

Future work This project does not take into account that comparing pairs of words ignoring the rest of the vocabulary can lead to false positives. This is because even if there are couples of words that are lexically close, there may be other words that are even closer or that the speaker may recognize and resolve the disambiguation. For instance, the words *casa* [ˈkā.zə] and *case* [ˈkā.sē] (*house* in Catalan and a conjugation of *to marry* in Spanish) are not considered false friends, as the word

	0	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**	**1**
1	0	0	0	0	0	0	0	0	0	0	0
0.9	0	3	16	67	159	495	1259	2064	3132	3421	3421
0.8	10	163	525	947	1450	2398	4051	5623	7755	8358	8358
0.7	171	2094	5531	7919	9683	12102	15178	17608	20498	21258	21258
0.6	966	9222	21815	28815	32800	36876	41205	44292	47526	48346	48346
0.5	4205	34331	74641	92614	100297	106106	111412	114805	118232	119094	119094
0.4	44195	307135	592481	688619	713498	723485	730173	734053	737679	738562	738562
0.3	142042	903329	1647083	1871525	1920216	1934274	1941692	1945725	1949410	1950298	1950298
0.2	306833	1878220	3311409	3707306	3783962	3802089	3810139	3814302	3818021	3818911	3818911
0.1	508243	3057321	5275462	5848466	5951185	5972925	5981520	5985781	5989525	5990419	5990419
0	534010	3189138	5482143	6071345	6176405	6198463	6207109	6211373	6215118	6216012	6216012

Table 2: Baseline model. Normalized spelling edit similarity $l \in 1..0$ over embeddings cosine similarity $c \in 0..1$. Each entry counts the frequency number of pairs of words with normalized spelling edit similarity $> l$ and cosine similarity $< c$.

	0	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**	**1**
1	0	0	0	0	0	0	0	0	0	0	0
0.9	0	3	16	65	155	482	1233	2014	3051	3322	3322
0.8	10	159	513	905	1373	2273	3832	5327	7347	7911	7911
0.7	166	1994	5272	7460	8998	11137	13859	16099	18793	19482	19482
0.6	804	7824	18645	24274	27304	30548	34140	36856	39816	40541	40541
0.5	2791	24703	54282	66539	71462	75591	79737	82644	85751	86504	86504
0.4	19884	157871	313470	362427	374291	380198	385071	388301	391559	392325	392325
0.3	36358	282931	548013	625039	641375	648128	653206	656492	659772	660541	660541
0.2	52604	399835	762592	863586	883610	890938	896126	899438	902731	903501	903501
0.1	77513	580054	1086307	1219611	1244238	1252272	1257599	1260946	1264252	1265022	1265022
0	77531	580120	1086392	1219704	1244331	1252365	1257692	1261039	1264345	1265115	1265115

Table 3: Final model. Same as Table 2 but adding the constraint of having a weighted phonetic distance smaller than 5.

casa also exists, with the same meaning of *word* in Spanish also with a close pronunciation: ['kā.sā].

Another improvement is fixing the weights in the weighted phonetic edit distance. Vowels could be fine-tuned to concrete pairs of languages, taking special account of the present vowels in the language. For instance, the /ə/ sound does not exist in many Romance languages and it is still considered as part of the graph regardless of the analyzed language. The same applies for consonants: here they are all considered as one big group, while in reality they can be further categorized by *manner* and *place* of articulation.

Also, many false positives are words that are not part of the language but are still included in the datasets because they appear in the text dumps. They could be avoided by preprocessing.

Finally, due to time constraints, we could just use the Spanish and the Catalan datasets. More pairs of languages have to be analyzed: first Romance languages and then languages from different groups (e.g., Germanic languages or Slavic languages).

References

Biennale and R. LeBlanc. 1989. *L'enseignement des langues secondes aux adultes: recherches et pratiques : communications présentées à la première Biennale de l'Institut des langues vivantes, Université d'Ottawa*. Actexpress Series. Presses de l'Université d'Ottawa.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

P. Boyd-Bowman. 1980. *From Latin to Romance in Sound Charts*. From Latin to Romance in Sound Charts. Georgetown University Press.

P.J. Chamizo-Domínguez. 2012. *Semantics and Pragmatics of False Friends*. Routledge Studies in Linguistics. Taylor & Francis.

Karl Pearson F.R.S. 1901. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572.

Oana Frunza and Diana Inkpen. 2009. Identification and disambiguation of cognates, false friends, and partial cognates using machine learning techniques. *International Journal of Linguistics*, 1(1):1–37.

Susan M. Gass. 1988. Second language vocabulary acquisition. *Annual Review of Applied Linguistics*, 9:92–106.

International Phonetic Association. 1999. *Handbook of the International Phonetic Association: A Guide to the Use of the International Phonetic Alphabet*. Cambridge University Press.

Armand Joulin, Piotr Bojanowski, Tomas Mikolov, Hervé Jégou, and Edouard Grave. 2018. Loss in translation: Learning bilingual word mapping with a retrieval criterion. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

P. Ladefoged and K. Johnson. 2010. *A Course in Phonetics*. Cengage Learning.

Robert Lado. 1957. *Linguistics across cultures: Applied linguistics for language teachers*. Univ of Michigan Pr.

Vladimir Iosifovich Levenshtein. 1966. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, 10(8):707–710. Doklady Akademii Nauk SSSR, V163 No4 845-848 1965.

Paul Nation. 2003. The role of the first language in foreign language learning. *Asian EFL journal*, 5(2):1–8.

Agnieszka Otwinowska and Jakub M. Szewczyk. 2019. The more similar the better? factors in learning cognates, false cognates and non-cognate words. *International Journal of Bilingual Education and Bilingualism*, 22(8):974–991.

Derek Rogers and Luciana d'Arcangeli. 2004. Italian. *Journal of the International Phonetic Association*, 34(1):117–121.

J.S. Vilar. 2009. *Millorem la pronúncia*. Col·lecció Recerca. Acadèmia Valenciana de la Llengua.

Martijn Wieling, Jelena Prokić, and John Nerbonne. 2009. Evaluating the pairwise string alignment of pronunciations.