Adaptive Complex Word Identification through False Friend Detection

Alessio Palmero Aprosio
Fondazione Bruno Kessler
Trento, Italy
aprosio@fbk.eu

Stefano Menini
Fondazione Bruno Kessler
Trento, Italy
menini@fbk.eu

Sara Tonelli
Fondazione Bruno Kessler
Trento, Italy
satonelli@fbk.eu

ABSTRACT

Automated complex word identification (CWI) is a crucial task in several applications, from readability assessment to lexical simplification. So far, several works have modeled CWI with the goal of targeting the needs of non-native speakers. However, studies in language acquisition show that different native languages can create positive or negative interferences w.r.t. reading comprehension, favouring or hindering the understanding of a document in a foreign language. Therefore, we propose to modify CWI to address the specific difficulties connected to different native languages. In particular, we present a pipeline that, based on the user native language, identifies complex terms by automatically detecting cognates and false friends on the fly. The selection presented by the CWI module is adaptive in that it changes depending on the native language of the user.

We implement and evaluate our approach for four different native languages (French, English, German and Spanish), in a setting where documents are written in Italian and should be read by language learners with low proficiency. We show that a personalised strategy based on false friend detection identifies complex terms that are different from those usually selected with standard approaches based on word frequency.

CCS CONCEPTS

• **Information systems** → **Content analysis and feature selection**; • **Applied computing** → *Document analysis.*

KEYWORDS

Complex word identification, text simplification, false friend detection, L2 speakers

ACM Reference Format:
Alessio Palmero Aprosio, Stefano Menini, and Sara Tonelli. 2020. Adaptive Complex Word Identification through False Friend Detection. In *Proceedings of the 28th ACM Conference on User Modeling, Adaptation and Personalization (UMAP '20), July 14–17, 2020, Genoa, Italy.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3340631.3394857

UMAP '20, July 14–17, 2020, Genoa, Italy

ACM ISBN 978-1-4503-6861-2/20/07…$15.00
https://doi.org/10.1145/3340631.3394857

1 INTRODUCTION

Automated complex word identification (CWI) [21], i.e. the prediction of which words challenge a target given population, is a crucial task in several applications, from readability assessment to text simplification. It has gained increasing attention in recent years, starting from the first shared task organised in 2016 at SemEval [18], in which it was introduced as an evaluation exercise to identify which words challenge non-native English speakers. Although the task organisers found interesting correlations between number of complex words manually identified by annotators, and their age and English proficiency, no analysis has been carried out concerning whether there are differences in the choice of complex words depending on the annotators' native language.

A second shared task has been organised in 2018 [32], extending the exercise to German, Spanish and French. In order to further investigate the different complexity perceived by native and non-native speakers, each term was annotated as complex or not by native and non-native speakers. An analysis of the released dataset[1] shows that indeed there are relevant differences between the two groups of annotators for each of the considered languages. For example, while 5,040 English terms were considered complex by both native and non-native speakers, 3,440 were hard to understand only for native speakers and 2,809 only for non-native ones. Similar differences can be observed for all the languages included in the task. This suggests that one-fits-all solutions to complex word identification are not reliable, and that the user native language should be taken into account in the task. Furthermore, the task organisers report that native annotators agree more within their group than non-native speakers, suggesting that inside the second group there are more diverse CWI needs.

Based on these previous observations, we propose to further refine the framework for complex word identification, moving towards a personalised setting, in which complex words are identified based on the specific native language of the user. Our approach is built upon research on the impact of *false friends* and *cognates* on language learners. The concept of false friends has been used to designate those pairs of words in two different languages that are similar in form but semantically divergent [7], like for example the English word *library* and the French one *librairie* (bookshop). On the contrary, words with the same meaning and a similar spelling in two or more languages are called cognates (e.g. English *colour* and French *couleur*). In multilingual settings, for instance while teaching, learning or translating a foreign language, these two phenomena have proven to be very relevant [20], because the lexical similarities between the two languages in contact have proven to

[1] Available at http://bit.ly/comp-word-id-ds

create interferences, favouring or hindering the course of learning. In the first case, lexical similarities can facilitate the understanding of a new language through so-called *positive transfer*. In the second case, false friends can cause misunderstandings because of *negative transfer*.

In the light of these findings, we consider for the first time the role of false friends in the task of complex word identification, and we implement an approach to identify complex words for non-native speakers, able to adapt to the user mother tongue. Our basic assumption is that all false friends are to be considered complex words for non-native language speakers, while cognates are considered easy to understand. This has been confirmed in several works on language acquisition and adult learners of a second language, especially in the early stages of acquisition [24, 26]. However, it has never been implemented in a Natural Language Processing pipeline for complex word identification.

The main contributions of this work are the following:

- We introduce the novel task of personalised complex word detection adapting to the mother tongue of non-native speakers and based on false friend identification
- We present and evaluate a complete pipeline to deal with the task
- We investigate the effectiveness of multilingual word embeddings [14] on false friend detection for four language pairs with different degrees of typologically similarity
- We create manually curated datasets of cognates and false friends for these language pairs and we make them freely available for future studies, together with the code implemented for the experiments

Our approach has been designed for native speakers of English, French, German and Spanish who want to read a document in Italian, extending the approach addressing only French speakers presented in [1].

The paper is structured as follows. In Section 2 we summarise the state of the art concerning false friend detection and complex word identification. In Section 3 we describe the pipeline we implemented for complex word identification based on false friend detection, including the pre-processing step of candidate selection. Then, in Section 4 the datasets developed for the experiments are described, together with the experimental setup for the two tasks of candidate selection and false friend detection. In Section 5 we present the results of our evaluation, concerning both the classification task and the comparison between different complex word identification strategies. Finally, pending issues and future extensions of this work are discussed in Section 6.

2 RELATED WORK

Our work is mainly related to two research areas in NLP: the one on *false friend / cognate identification*, which lies at the core of our personalised approach, and the task of *complex word identification* (CWI), in which we propose to integrate the first task.

As for false friend and cognate detection, several works in the past have used orthographic and phonetic similarity to detect them [10, 11, 16, 27], while the use of semantic information is a less-studied problem. The intuition behind this approach is that two cognates have a shared semantics, as opposed to false friends, whose meanings are generally unrelated. The semantic distance between candidate pairs has been measured in two main ways: using taxonomic similarity based on WordNet [15], or using distributional similarity [13, 15], based on the assumption that cognates should be semantically close to the same set of words in two or more languages. In order to compute the above semantic distance, candidate pairs have been mapped to a monolingual setting using different approaches, including the use of bilingual dictionaries [15] and the alignment of parallel bi-texts and web snippets [17]. In our approach, instead, we adopt a multilingual embedding space [14], so that we can measure the semantic distance between the candidate pairs in their original language. While the use of monolingual word embeddings for cognate identification has already been proposed in few works in the past [23], we identified only two previous works performing false friend detection based on multilingual embeddings, namely Castro et al. [4] and Uban et al. [28]. Compared to [4], our work is different, since we use different types of embeddings (Fasttext vs. word2vec), we perform our evaluation on four language pairs rather than only one, and we also add a candidate selection part evaluating different similarity metrics (see Section 3.1). Compared to [28], we use similarity metrics obtained from a multilingual embedding space as features in a supervised setting, while Uban et al. consider a pair of candidate terms to be false friends if in the shared semantic space, there exists a word in the second language which is semantically closer to the original word than its candidate cognate. In other words, they simply use cosine similarity to identify cognates in an unsupervised setting.

We cast false friend and cognate detection as a necessary step to adapt the identification of complex words to the user's native language. Past research so far has distinguished between complex word identification for native and non-native speakers [18, 32] but differences among specific native languages have not been investigated yet, even if the robustness of cross-lingual models for CWI has been tested [5]. In both shared tasks for CWI, the best performing systems [6, 19] use supervised approaches relying on a number of morphological, lexical, collocational and semantic features. The organisers of 2018 task confirm that complex words are generally longer than simple ones, but also that systems have a bias against short terms, since they tend to classify them as simple, leading to a high number of false negatives [32]. In our approach, instead, we include only semantic information related to the terms and not their surface information, without considering word length. Other recent works have devised approaches to identify complex words tailored to users' proficiency in a foreign language, after asking 15 Japanese native speakers to manually evaluate their knowledge of 12,000 English terms and building a personalised model based on this knowledge [12]. The same approach has then been applied to the selection of reading material matching the proficiency level of learners in an adaptive way [31]. However, none of the above works has tackled the automated detection of false friends to be integrated in a CWI system, and the personalisation has pertained the proficiency of single users, not the learners' native language.

3 GENERAL FRAMEWORK

The complete pipeline that we propose for personalised complex word identification is displayed in Figure 1: the modules on top of

the diagram are run in real time given a document provided by a user who is a non-native language speaker. The modules with a light blue background, instead, are run once to pre-compute the resources needed for the CWI task.

The first stage involves the extraction of false friend/cognate candidates given a corpus in input and a list of monolingual terms (dictionary list) in another language (the user's mother tongue), whereas the second stage is concerned with the classification of the extracted pairs as cognates or false friends. The extraction of candidate pairs is based on orthographic similarity between two words, whereas the classification of the extracted pairs is performed on the basis of their semantic similarity in a supervised fashion.

More formally, given a document D_i written in the language L_I, and a native language L_N spoken by a user, our goal is to identify in D_i all words that the user may consider complex (i.e. false friends), discarding those that s/he can easily understand (i.e. cognates). The approach consists of the three following steps:

(1) **Candidate selection**: for each content word w_i in D_i, we automatically identify a list of words $W_N \subset L_N$ which are orthographically similar to w_i, and that may be either cognates or false friends. In this phase, several orthographical similarity metrics are evaluated.
(2) **False friend and cognate detection**: for the most similar words of w_i in W_N, we classify whether they are false friends of w_i or not.
(3) **Complex word identification**: Based on the output of the previous setup, the system identifies w_i as a complex word or not for native speakers of L_N.

Each step is further detailed the following Subsections.

3.1 Candidate Selection

In order to get the word (or words) in a language L_N that are most similar to an input word w_i, we rely of the list of lemmas found in a dictionary of L_N. Since several dictionaries are available in electronic format, this kind of resource is quite easy to obtain.

A number of similarity metrics have been presented in the past to identify candidate cognates and false friends, see for example the evaluation in [9]. We choose three of them, motivated by the fact that we want to have at least one n-gram based metric (XXDICE) and one non-ngram based (Jaro/Winkler). To that, we add a more standard metric, Normalized Edit Distance (NED). The three metrics are explained below:

- XXDICE [3]: It takes in consideration the shared number of extended bigrams[2] and their position relative to two strings S_I and S_N. The formula is:

$$\mathrm{XX}(S_I, S_N) = \frac{\sum_B \frac{2}{1+(\mathrm{pos}(x)-\mathrm{pos}(y))^2}}{\mathrm{xb}(S_I) + \mathrm{xb}(S_N)}$$

  where B is the set of pairs of shared extended bigrams (x, y), x in S_I and y in S_N. The functions $\mathrm{pos}(x)$ and $\mathrm{xb}(S)$ return the position of extended bigram x and the number of extended bigrams in string S respectively.

[2] An extended bigram is an ordered letter pair formed by deleting the middle letter from any three letter substring of the word.

- NED, Normalized Edit Distance [29]: A regular Edit Distance calculates the orthographic difference between two strings assigning a cost to any minimum number of edit operations (deletion, substitution and insertion, all with cost of 1) needed to make them equal. NED is obtained by dividing the edit cost by the length of the longest string.
- Jaro/Winkler [30]. The Jaro metric for two strings S_I and S_N is computed as follows:

$$J(S_I, S_N) = \frac{1}{3} \cdot \left(\frac{m}{|S_I|} + \frac{m}{|S_N|} + \frac{m-T}{m} \right)$$

  where m is the number of characters in common, provided that they occur in the same (not interrupted) sequence, and T is the number of transpositions of character in S_I to obtain S_N. The Winkler variation of the metric adds a bias if the two strings share a prefix:

$$JW(S_I, S_N) = J(S_I, S_N) + (1 - J(S_I, S_N))lp$$

  where l is the number of characters of the common prefix of the two strings, up to four, and p is a scaling factor, usually set to 0.1.

Each of the three measures has some disadvantages. For example, we found that Jaro/Winkler metric boosts the similarity of words with the same root. On the other hand, with NED we extract several pairs of words having the same similarity score. As a result, two words that are close according to a metric can be very different using another metric. To overcome this limitation, we balance the three metrics by computing a coefficient for each of the three scores, tuning them on a training set. For details, see Section 'Experimental Setup'. The coefficients significantly reduce the number of pairs of words having the same similarity score and allow, in most of the cases, the identification of a single pair, whose similarity is higher than all the others.

3.2 False Friend and Cognate Detection

After identifying the term (or terms) in L_N that are most similar to the input word w_i, an SVM-based classifier is trained to detect if they are false friends and therefore can be considered as complex words according to the native language of the reader. In order to train the classification model, the following data are used: *i)* multilingual embeddings where L_I and L_N are aligned and *ii)* a synonym list for L_I. A list of bilingual terms in L_I and L_N manually labelled as false friends or cognates is needed for training and testing. Given a bilingual word pair w_i and w_n to be classified, we extract 5 features to train the classification model:

Cosine similarity: we use the vectors of w_i and w_n from the respective aligned semantic spaces to compute their cosine distance as a measure of their semantic similarity. The intuition behind this approach is that two cognates have a shared semantics and therefore a high cosine similarity, as opposed to false friends, whose meanings are generally unrelated.

Cosine with synonyms (cross-lingual, mean value): using a synonym list for L_I, we identify all synonyms of w_i.Then, for each synonym of w_i we compute its cosine distance with w_n in L_N. We use the means of the cosine of all synonyms as feature.

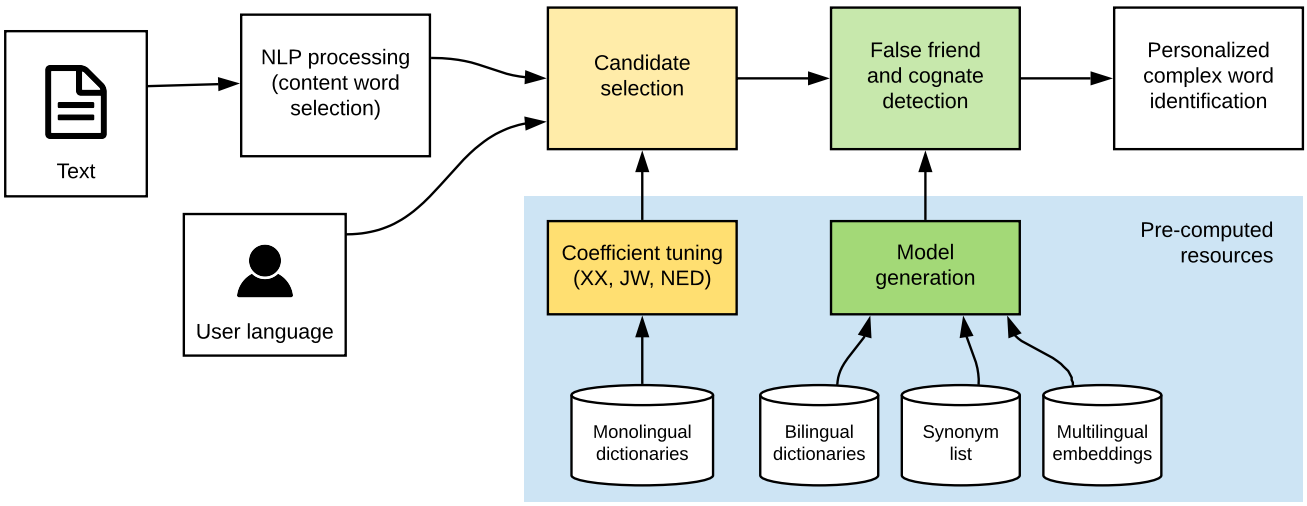

Figure 1: The pipeline for personalised complex word identification based on the user native language

Cosine with synonyms (cross-lingual, max value): as in the previous feature we use the synonym list to calculate the cosine distance between every synonym of w_i and w_n. As feature, we consider the highest cosine obtained among all the synonyms of w_i.

Cosine best synonym - monolingual: we compute the cosine distance between w_i in L_I and its synonyms, and we use as feature the cosine of the closest one.

Cosine best synonym - cross-lingual: as in the previous feature, we select the closest synonym to w_i. Then, we compute its cosine distance with w_n.

We also consider two baselines: a random classifier, weighted on the distribution of cognates and false friends; a concatenation of the embeddings of w_i and w_n in the multilingual space.

3.3 Complex Word Identification

The last step is to adopt the approach described in the previous subsections to recognize the presence of complex words in a text read by a non-native speaker. Starting from a text in L_I, we first lemmatize it and then we consider for each content word w_i the classifier output: if w_i is a false friend, then it is labeled as complex for non-native speakers having L_2 as mother tongue.

If several candidate pairs are passed to the classifier, because more than one term w_n shows the same similarity to w_i, then at least one of them should be classified as false friend to identify w_n as a complex word.

4 EXPERIMENTAL SETUP

In our experiments, we consider a setting in which Italian documents are read by users with different native languages: German, English, Spanish and French. Each step is therefore evaluated on four language pairs.

4.1 Datasets

For training and testing, four datasets were manually created by a linguist with experience in L2 teaching and then manually checked by another domain expert. Each dataset contains pairs of orthographically similar words marked as either cognates or false friends. These terms were collected from several lists available on the web including online dictionaries and teaching material, as well as scientific papers on false friends and L2 learning. The annotator validated them manually and enriched, when needed. Table 1 shows the number of cognates and false friends in each dataset. The data are then split in a training and a test set (around 85% and 15% respectively).

Table 1: Statistics on the four datasets.

Lang. pair	Cognates	False friends
Ita-En	960	1,144
Ita-Fr	940	591
Ita-De	466	170
Ita-Sp	523	384

The four datasets are available for download, together with the code implemented to run the experiments.[3]

4.2 Experimental Setup for Candidate Selection

The goal of the candidate selection step is to obtain for each term w_i in Italian, the term(s) with the highest orthographic similarity in the language of a non-native speaker (English, German, Spanish or French).

The list of lemmas for each language are extracted from an electronic dictionary[4] and then used to compute their similarity

[3] https://github.com/dhfbk/falsefriends
[4] https://www.collinsdictionary.com/. Only single words are considered, not multiwords.

with w_i. The lemmas are normalized for accents and diacritics, in order to avoid poor results of the metrics when dealing with French, Spanish and German words. For example, in cases like *général* and *generale*, the accented *é* character would be considered different with respect to *e*.[5]

In order to compute the best way to balance the three similarity metrics detailed above, we compute all the possible combinations of coefficients between the three measures for all the word pairs in the four training sets and then keep the combination that scores the highest similarity. Note that, since in this step the goal is to find candidates with high orthographic similarity to w_i, both cognates and false friends in the training set are considered positive examples. The coefficients α, β and γ that are applied to the metrics are defined so that $\alpha+\beta+\gamma = 1$ and $\alpha, \beta, \gamma \in \{0, 0.1, 0.2, \ldots, 0.9, 1\}$ (e.g *0.1-0.1-0.8* and *0.1-0.2-0.7*).

Since in our setting the most similar result in the ranking is the only relevant element we want to retrieve, we estimate the different coefficients by computing Precision at 1 (P@1). We search for the best triple of coefficients (α, β, γ), so that the resulting metric

$$\alpha m_{\text{XX}} + \beta m_{\text{JW}} + \gamma m_{\text{NED}}$$

maximizes P@1, where m_{XX}, m_{JW}, and m_{NED} are the values of the three metrics XXDICE (XX), Jaro/Winkler (JW), and Normalized Edit Distance (NED) respectively.

Results are reported in Table 2. We first present P@1 obtained with the three different scores in isolation, and then in the last row the best combination of coefficients. We also report in the Table the Mean Reciprocal Rank (MRR) value for comparison.

While for each language we obtain a set of configurations scoring very similar results, with differences that are not statistically significant, we observe that among them the values $(\alpha, \beta, \gamma) = (0.1, 0.4, 0.5)$ were the best for all the languages (see the last row of the table). This configuration is the best performing one over the different languages, therefore it is the one that will be used in the next experimental steps.

4.3 Experimental Setup for False Friend detection

To calculate the features used by the SVM classifier, in particular the cosine similarity between pairs of words in different languages, we use the embeddings from [8] trained on Wikipedia data[6]. We choose these resources because they are available for all the languages in our experiments, as well as several other languages.

In order to create a shared multilingual embedding space, it is necessary to align the monolingual embeddings of English, French, German, Spanish to the Italian one, being the language of the documents where we want to identify complex words. We implement the approach presented in [22] that, starting from a bilingual training dictionary, learns a linear transformation to place the languages in a single space. The lists of word pairs w_i (Italian) and w_n (secondary language) are extracted from the online dictionaries available at http://dizionari.corriere.it/ and contain 132,130 word pairs for Italian-English, 25,017 for Italian-French, 127,904 for Italian-German and 23,683 for Italian-Spanish.

As for the synonym list required to compute part of the cosine-based features, we obtain it from the online version of *Dizionario dei Sinonimi e dei Contrari*.[7] Note that this list is necessary only for the language that non-native users aim to learn, not for all the languages involved in the experiments.

To find the optimal configuration of kernel, parameter, and features, we run the SVM classifier in a 10-fold cross-validation on the training set for every language pair. Results are displayed in Table 3. The best performance is obtained with polynomial kernel of degree 3, C parameter = 100 and epsilon = 0.5 and by using all the features. However, by ablating the four features based on the synonym list (*Only cosine similarity*), we observe only a limited drop in accuracy, showing that our approach may work well also if a synonym list is not available for the language of interest. Indeed, only a bilingual list for aligning the multilingual space is really necessary to perform the experiments for additional languages.

5 EVALUATION

We perform two types of evaluation: the first one is aimed at assessing the performance of the false friend classifier on the test set with the best configuration obtained on the training set in 10-fold cross-validation. The second one is more general-purpose, and is aimed at assessing the impact of the personalised complex word detection workflow presented in this paper, compared with a standard approach based on a pre-defined list of simple words in Italian.

As for the first evaluation, results are reported in Table 4, obtained with *All features* as reported in Table 3. Precision, Recall and F1 are computed on the false friend class, while Accuracy takes into account all correctly classified pairs (both cognates and false friends).

This first evaluation shows that the size of the training sets has a limited impact on classifier performance. In particular, while most training instances are available for the Italian–English pair, the corresponding classifier achieves the worst performance in both settings in terms of accuracy. On the contrary, the Italian–Spanish dataset contains less than 1,000 pairs, but classification yields the best scores, as shown both in Table 3 and 4. We may therefore argue that typologically similar language pairs are easier to classify that less related ones. This finding is in contrast with what was observed in [15], in which classification results on English-Spanish were better than on French-Spanish. While the authors claimed that this outcome was probably affected by the corpora used for distributional similarity measures and by the quality of semantic resources used, in our case these factors do not come into play. Indeed, the Italian-Spanish and Italian-French pairs achieve the highest classification results in the 10-fold cross-validation setting, which suggests that this kind of task performed among Romance languages may be easier than between a Romance and a Germanic language.

The second evaluation is aimed at assessing the impact of the personalised approach to CWI presented in this paper compared with a more standard approach based on pre-defined lists. Our goal is to check whether different strategies, and in particular different native languages set by users, lead to different CWI choices.

[5] For example, NED between *général* and *generale* returns 0.375 when the two strings are not normalized and 0.125 when they are.
[6] Available at http://bit.ly/fasttext-vectors
[7] http://bit.ly/sin-contr

Table 2: Analysis of the English, French, Spanish, German candidate selection strategy using different metrics

Coefficient value			English		French		Spanish		German	
α	β	γ	**MRR**	**P@1**	**MRR**	**P@1**	**MRR**	**P@1**	**MRR**	**P@1**
0	0	1.0	0.56	44.19	0.50	37.16	0.85	78.46	0.75	65.47
0	1.0	0	0.63	46.00	0.57	37.16	0.87	76.96	0.77	58.32
1.0	0	0	0.65	48.23	0.58	38.01	0.87	71.79	0.77	60.63
0.1	0.4	0.5	0.64	**55.56**	0.58	**47.87**	0.88	**85.48**	0.78	**72.21**

Table 3: Results on Italian-English/French/German/Spanish training pairs using 10-fold cross-validation.

	Accuracy (%)			
Features	**German**	**English**	**Spanish**	**French**
Random Baseline	61.06	51.43	53.49	54.20
Concatenated Embeddings	81.85	75.49	73.70	77.18
Only cosine similarity	86.72	78.89	91.93	88.97
All cosine-based features - no embeddings	85.42	80.92	92.18	89.35
All Features (all cosine-based + embeddings)	**87.52**	**81.26**	**93.01**	**90.69**

Table 4: Results of the classification on testsets. P/R/F1 on false friend classification, Accuracy on false friend and cognate classification

	Precision	**Recall**	**F1**	**Accuracy** (%)
English	0.86	0.86	0.86	84.45
French	0.88	0.76	0.81	84.85
Spanish	0.88	0.91	0.89	89.47
German	0.86	0.68	0.76	87.23

We consider as a comparison De Mauro's lexicon of Basic Italian[8], the de-facto standard resource for studies on lexical complexity in Italian. This lexicon contains around 7,500 terms extracted from documents representative of different textual genres and manually evaluated to include only lemmas that are deemed 'easy to read' by experts in sociolinguistics. The resource is further split into three lists: the 2,000 terms belonging to the 'fundamental Italian vocabulary', the 3,000 terms being 'most frequently used' and the 2,500 considered 'highly familiar'. We randomly collect 500K sentences from the multi-domain Italian corpus described in [25]. Next, we remove all sentences containing at least one word from our training sets, resulting in a set of 24,028 sentences. We then lemmatise and extract all content words using the TINT NLP Suite [2]. We then identify these words as being complex (or not) using different strategies:

(1) **Standard approaches**: A term is complex if it is not included in the Lexicon of Basic Italian (LBI), or it is not included among the 'fundamental' and 'most frequently used' terms in the Lexicon of Basic Italian (LBI-small)
(2) **Personalised approaches:** A term is complex if our pipeline classifies it as a false friend for English native speakers (EN), for German native speakers (DE), for French native speakers (FR), for Spanish native speakers (ES)

[8]https://www.dropbox.com/s/mkcyo53m15ktbnp/nuovovocabolariodibase.pdf?dl=0

Table 5: Number of tokens marked as complex in the 24,028 test sentences for different user native languages. In brackets, the number of types.

Native language	**N. identified complex words**	
EN	20,486	(1,569)
DE	11,634	(544)
FR	21,104	(1,521)
ES	12,223	(949)
LBI	22,209	(7,352)
LBI-small	32,412	(9,021)

We finally compare the output obtained with the different strategies, measuring the number of identified complex words using the standard and the personalised approaches. For the second setting, we proceed as follows: given the lemma of a content word w_i, we identify the most similar word in the language of the user based on the $(0.1, 0.4, 0.5)$ similarity metric coefficients introduced above. In case of several first-ranked words having the same similarity, we extend the selection to all the candidates. Each word pair including w_i and the candidate(s) is then classified as false friend or not using the SVM classifier specifically trained for the user language. Statistics are reported in Table 5.

Results show that an approach considering as complex all words that are not listed as 'fundamental' and 'most frequently used' in the Lexicon of Basic Italian (LBI-small) tends to identify a large number of complex words, and that more generally the vocabulary-based approaches are less conservative. A strategy targeting the language of Italian non-native speakers, instead, seems to be more

Table 6: Number of tokens marked as complex in the test sentences for different user languages and approaches. In brackets, the number of types.

	EN		DE		FR		ES	
EN	-		6,961	(361)	10,646	(767)	5,489	(439)
DE	6,961	(361)	-		7,185	(340)	3,505	(189)
FR	10,646	(767)	7,185	(340)	-		5,893	(435)
ES	5,489	(439)	3,505	(189)	5,893	(435)	-	
LBI	3,653	(898)	1,452	(285)	3,328	(801)	4,479	(588)
LBI-small	5,647	(1,248)	1,953	(403)	5,222	(1,190)	5,341	(793)

focused. We further analyse the results by checking whether there are overlaps among the approaches, i.e. if our pipeline identifies as complex the same terms present in LBI/LBI-small, and if different user languages lead to the identification of different complex words. Results are reported in Table 6. We observe that the standard and the personalised approaches show some overlaps, i.e. some false friends have been included also in the LBI and LBI-small lists. However, more overlaps are found among the outputs targeting different user languages. CWI for French native speakers leads to the identification of most complex words among the four available languages (21,104), and also to the highest number of complex words shared with other languages. In the personalised setting, neither the size of the datasets used to train the false friend classifier, nor typological differences among the languages seem to play a relevant role in the CWI overlaps. The different outcome of the standard and the personalised approaches is confirmed also if we consider the average length of the Italian words identified as complex: using LBI it is 8.67 characters/type, while in the language-specific setting the average length is always lower (EN=7.26, DE=5.92, FR=7.12, ES=7.08). This shows that in a personalised setting the bias against short terms, mentioned in the *Related work* section, is limited.

In order to better understand the cases of overlap between languages, we manually inspected the pipeline output. As an example, we report in Table 7 the different CWI strategies on two test sentences. The tokens in bold are complex words identified based on LBI-small list. Since this resource has been created taking mainly frequency and usage into account, 'omicidio' in the first example is considered difficult to understand. However, it is probably easy for English speakers, since its translation 'homicide' is also a cognate of the Italian term, and for this reason it is not identified by the CWI pipeline targeting English speakers. On the contrary, 'rende' is very frequent in Italian and it is therefore excluded from LBI-small. However, it is a false friend of 'to render', which makes it a complex term for English native speakers. In the second example, 'Attualmente' is recognised as complex both in the LBI-small setting and by the CWI pipeline targeting English speakers. Indeed, it is a rather unusual adverb in Italian, but also a false friend of 'actually' in English.

Table 7: Example sentences with different CWI strategies. LBI-small output in bold, language-specific complex words between brackets.

Capisco che lei si $\lfloor$senta$\rfloor_{es,fr}$ **responsabile** per ciò: che è accaduto a suo fratello , ma ciò non la $\lfloor$rende$\rfloor_{en}$ **colpevole** di **omicidio**.
I understand that you feel responsible for what happened to your brother, but this does not make you responsible for a homicide.
$\lfloor$**Attualmente**$\rfloor_{en}$, gli U2 si stanno $\lfloor$prendendo$\rfloor_{es}$ una **pausa** dalla **tournè**
U2 are currently taking a break from their tour.

6 CONCLUSIONS AND FUTURE WORK

In this work, we presented a novel integrated pipeline to identify complex words for language learners adapting to the native language of the users. We describe a supervised approach to automatically detect false friends in four language pairs that relies on a small set of features easily obtained from multilingual embeddings. We also evaluate the best combination of coefficients to combine different similarity metrics when selecting false friend candidates.

The pipeline that we propose is implemented with the goal of selecting complex words in Italian texts according to a strategy able to adapt to the mother tongue of non-native speakers. We experiment with four different language pairs, but the results suggest that the approach can be extended with limited resources also to other languages, at least European ones. Indeed, multilingual embeddings are available for many languages and lists of cognates and false friends, although of limited size, can be easily retrieved online.[9] Although the measures of orthographic similarity used for candidate selection (Section 3.1) assume that the two languages share the same writing system, these could be replaced by measures of phonetic similarity in case of different alphabets. Our results suggest however that the approach works better for typologically similar languages.

The evaluation of the impact of this personalised strategy compared with a standard frequency-based resource shows that standard approaches tend to identify many more words as complex and to favour longer terms, beside avoiding any form of personalisation. Instead, the language-specific choices suggested by our pipeline are not limited to a subset of the frequent complex words, but are driven by different word features.

The pipeline has been integrated in an online system that performs lexical simplification, in which the words to be simplified are

[9] See for example the Wiktionary entries at http://bit.ly/wiktionary-ff

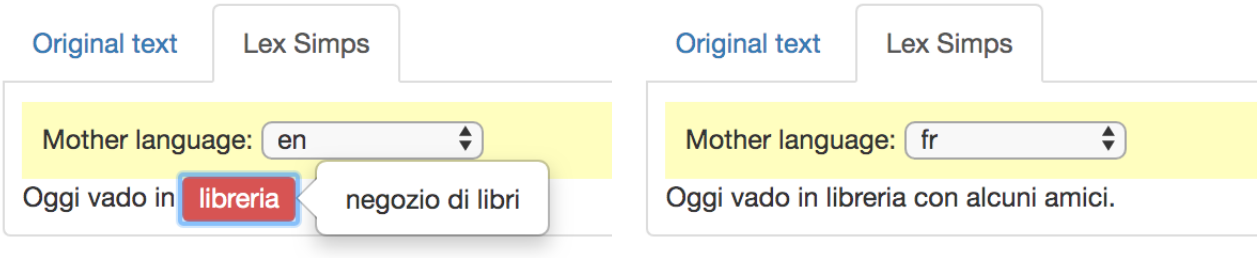

Figure 2: Screenshot of the system for lexical simplification. The term 'libreria' (bookshop) is simplified only if the user is an English native speaker. No simplification is performed if French is selected as mother tongue, due to the presence of the cognate term 'librairie'.

selected based on the user mother tongue. In Figure 2 two screenshots are reported, showing the different system behaviour if the user selects English or French as mother language. In the first case, the term 'libreria' is automatically highlighted as difficult because the classifier found 'library' as a false friend. A simpler synonym ('shop where books are sold') is displayed, which is automatically selected from a synonym list ranked by frequency. In the second case, when French is selected, no complex term is highlighted, since 'libreria' and 'librairie' are automatically identified as cognates by the pipeline. The system does this analysis in real time and displays the results in few milliseconds.

A limitation of our approach is that false friends are seen as such without considering the context in which they appear. However, in some cases, the sentence in which they occur could make the meaning of a false friend very easy to infer, avoiding the need to trigger the simplification suggestion. In some other cases, single words may have different meanings, being both false friends and cognates. For example, 'sentir' in Spanish can be a cognate of 'sentire' in Italian (to feel), but also a false friend in the expression 'lo siento', meaning 'I am sorry'. A more complex setting, taking into account the whole sentence and not just the single terms in isolation, would be needed to account for these cases.

In the future we plan to exploit this online system to involve users with different mother tongues in an evaluation of the pipeline, adding to the online interface the possibility to give a feedback or to select terms that the system should highlight. This information could be used to increase the amount of training data. Another possible evaluation could involve a comparison between the standard and the personalised complex word identification, again using the online system for a controlled experiment. Finally, we plan to extend our approach to other language pairs, comparing typologically similar and different languages at scale.

ACKNOWLEDGMENTS

This work has been supported by the European Commission project SIMPATICO (H2020-EURO-6-2015, grant number 692819). We would like to thank Francesca Fedrizzi for her help in creating the gold standard.

REFERENCES

[1] Alessio Palmero Aprosio, Stefano Menini, Sara Tonelli, Luca Ducceschi, and Leonardo Herzog. 2018. Towards Personalised Simplification based on L2 Learners' Native Language. In *Proceedings of the Fifth Italian Conference on Computational Linguistics (CLiC-it 2018), Torino, Italy, December 10-12, 2018.* http://ceur-ws.org/Vol-2253/paper44.pdf

[2] Alessio Palmero Aprosio and Giovanni Moretti. 2018. Tint 2.0: an All-inclusive Suite for NLP in Italian. In *Proceedings of the Fifth Italian Conference on Computational Linguistics (CLiC-it 2018), Torino, Italy, December 10-12, 2018.* http://ceur-ws.org/Vol-2253/paper58.pdf

[3] Chris Brew, David McKelvie, et al. 1996. Word-pair extraction for lexicography. In *Proceedings of the 2nd International Conference on New Methods in Language Processing*. 45–55.

[4] Santiago Castro, Jairo Bonanata, and Aiala Rosá. 2018. A High Coverage Method for Automatic False Friends Detection for Spanish and Portuguese. In *Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2018)*. Association for Computational Linguistics, Santa Fe, New Mexico, USA, 29–36. https://www.aclweb.org/anthology/W18-3903

[5] Pierre Finnimore, Elisabeth Fritzsch, Daniel King, Alison Sneyd, Aneeq Ur Rehman, Fernando Alva-Manchego, and Andreas Vlachos. 2019. Strong Baselines for Complex Word Identification across Multiple Languages. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, Minneapolis, Minnesota, 970–977. https://doi.org/10.18653/v1/N19-1102

[6] Sian Gooding and Ekaterina Kochmar. 2018. CAMB at CWI Shared Task 2018: Complex Word Identification with Ensemble-Based Voting. Association for Computational Linguistics, 184–194. http://aclweb.org/anthology/W18-0520

[7] Sylviane Granger and Helen Swallow. 1988. False Friends: a Kaleidoscope of Translation Difficulties. *Le Langage et l'Homme* 23, 2 (1988), 108–120.

[8] Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning Word Vectors for 157 Languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

[9] Diana Inkpen and Oana Frunza. 2005. Automatic Identification of Cognates and False Friends in French and English. In *Proceedings of RANLP*. 251–257.

[10] D. Inkpen, O. Frunza, and G. Kondrak. 2005. Automatic identification of cognates and false friends in French and English. In *Proceedings of the international conference on recent advances in natural language processing (RANLP' 05)*. 251–257.

[11] Grzegorz Kondrak. 2009. Identification of Cognates and Recurrent Sound Correspondences in Word Lists. *TAL* 50, 2 (2009), 201–235. http://atala.org/IMG/pdf/TAL-2009-50-2-08-Kondrak.pdf

[12] John Lee and Chak Yan Yeung. 2018. Personalizing Lexical Simplification. In *Proceedings of the 27th International Conference on Computational Linguistics*. Association for Computational Linguistics, Santa Fe, New Mexico, USA, 224–232. https://www.aclweb.org/anthology/C18-1019

[13] Nikola Ljubešić and Darja Fišer. 2013. Identifying false friends between closely related languages. Association for Computational Linguistics, 69–77. http://www.aclweb.org/anthology/W13-2411

[14] Tomas Mikolov, Quoc V Le, and Ilya Sutskever. 2013. Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168* (2013).

[15] Ruslan Mitkov, Viktor Pekar, Dimitar Blagoev, and Andrea Mulloni. 2007. Methods for extracting and classifying pairs of cognates and false friends. *Machine Translation* 21 (2007), 29–53.

[16] Andrea Mulloni and Viktor Pekar. 2006. Automatic Detection of Orthographics Cues for Cognate Recognition. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation, LREC 2006, Genoa, Italy, May 22-28, 2006.*, Nicoletta Calzolari, Khalid Choukri, Aldo Gangemi, Bente Maegaard, Joseph Mariani, Jan Odijk, and Daniel Tapias (Eds.). European Language Resources Association (ELRA), 2387–2390. http://www.lrec-conf.org/proceedings/lrec2006/pdf/676_pdf.pdf

[17] Svetlin Nakov, Preslav Nakov, and Elena Paskaleva. 2009. Unsupervised Extraction of False Friends from Parallel Bi-Texts Using the Web as a Corpus. In *RANLP*.

[18] Gustavo Paetzold and Lucia Specia. 2016. SemEval 2016 Task 11: Complex Word Identification. In *Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, San Diego, CA, USA, June 16-17, 2016*, Steven Bethard, Daniel M. Cer, Marine Carpuat, David Jurgens, Preslav Nakov, and Torsten Zesch (Eds.). The Association for Computer Linguistics, 560–569. http://aclweb.org/anthology/S/S16/S16-1085.pdf

[19] Gustavo Paetzold and Lucia Specia. 2016. SV000gg at SemEval-2016 Task 11: Heavy Gauge Complex Word Identification with System Voting. Association for Computational Linguistics, 969–974. https://doi.org/10.18653/v1/S16-1149

[20] H. Ringbom. 1986. Crosslinguistic Influence and the Foreign Language Learning Process. In *Crosslinguistic Influence in Second Language Acquisition*, E. Kellerman and Smith Sharwood M. (Eds.). Pergamon Press, New York.

[21] Matthew Shardlow. 2013. A Comparison of Techniques to Automatically Identify Complex Words. In *51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, Proceedings of the Student Research Workshop, 4-9 August 2013, Sofia, Bulgaria*. The Association for Computer Linguistics, 103–109. http://aclweb.org/anthology/P/P13/P13-3015.pdf

[22] Samuel L. Smith, David H.P. Turban, Steven Hamblin, and Nils Y. Hammerla. 2017. Offline bilingual word vectors, orthogonal transformations and the inverted softmax. *ICLR* (2017).

[23] Adam St Arnaud, David Beck, and Grzegorz Kondrak. 2017. Identifying Cognate Sets Across Dictionaries of Related Languages. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Copenhagen, Denmark, 2519–2528. https://www.aclweb.org/anthology/D17-1267

[24] A. Talamas, J. F. Kroll, and R. Dufour. 1999. From form to meaning: Stages in the acquisition of second-language vocabulary. *Bilingualism: Language and Cognition* 2 (1999), 45–58.

[25] S. Tonelli, A. Palmero Aprosio, and M. Mazzon. 2017. The Impact of Phrases on Italian Lexical Simplification. In *Proceedings of the Fourth Italian Conference on Computational Linguistics (CLiC-it 2017), Rome, Italy, December 11-13, 2017*. http://ceur-ws.org/Vol-2006/paper027.pdf

[26] Claudio Tonzar, Lorella Lotto, and Remo Job. 2009. L2 Vocabulary Acquisition in Children: Effects of Learning Method and Cognate Status. *Language Learning* 59, 3 (2009), 623–646. https://doi.org/10.1111/j.1467-9922.2009.00519.x arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-9922.2009.00519.x

[27] P. Turchin, I. Peiros, and Gell-Mann M. 2010. Analyzing genetic connections between languages by matching consonant classes. *Journal of Language Relationship* 3 (2010), 117–126.

[28] Ana Uban, Alina Maria Ciobanu, and Liviu P. Dinu. 2019. Studying Laws of Semantic Divergence across Languages using Cognate Sets. In *Proceedings of the 1st International Workshop on Computational Approaches to Historical Language Change*. Association for Computational Linguistics, Florence, Italy, 161–166. https://doi.org/10.18653/v1/W19-4720

[29] Robert A Wagner and Michael J Fischer. 1974. The string-to-string correction problem. *Journal of the ACM (JACM)* 21, 1 (1974), 168–173.

[30] William E. Winkler. 1990. String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage. In *Proceedings of the Section on Survey Research Methods of the American Statistical Association*. ERIC.

[31] Chak Yan Yeung and John Lee. 2018. Personalized Text Retrieval for Learners of Chinese as a Foreign Language. In *Proceedings of the 27th International Conference on Computational Linguistics*. Association for Computational Linguistics, Santa Fe, New Mexico, USA, 3448–3455. https://www.aclweb.org/anthology/C18-1292

[32] Seid Muhie Yimam, Chris Biemann, Shervin Malmasi, Gustavo H. Paetzold, Lucia Specia, Sanja Stajner, Anaïs Tack, and Marcos Zampieri. 2018. A Report on the Complex Word Identification Shared Task 2018. *CoRR* abs/1804.09132 (2018). arXiv:1804.09132 http://arxiv.org/abs/1804.09132