Regular Sound Changes for Cross-Language Information Retrieval

Michael P Oakes and Souvik Banerjee

Abstract

The aim of this project is the automatic conversion of query terms in one language into their equivalents in a second, historically related, language, so that documents in the second language can be retrieved. The method is to compile lists of regular sound changes which occur between related words of a language pair, and substitute these in the source language words to generate target language words. For example, if we know *b* in Italian often corresponds with a *v* in Spanish, an unaccented *o* in Italian with *ó* in Spanish, and a terminal *e* in Italian is replaced with a null in Spanish, we can construct the Spanish word *autómovil* (car) from the Italian *automobile*.

A bilingual word list or dictionary is needed at first to first discover the set of regular sound changes, but once this is known, there is no further need for a dictionary to look up individual query words. The method is language pair independent, as long as the two languages belong to the same language family, such as the Romance languages. Buckley et al. (2000) proposed a related method based on knowledge of regular orthographic changes between languages, when corresponding words in two languages are pronounced alike, but the method proposed here is novel in that it also incorporates regular, linguistically attested, sound changes between historically related languages.

1. First experimental run: cognates sought by edit distance

Buckley et al. (2000) wished to determine how effective cross-lingual information retrieval (CLIR) could be with a minimal amount of linguistic information. They made no use of dictionaries, but instead treated the English query words as “potentially mis-spelled French words”. In order to use a set of English query words to retrieve French documents, they first stored all the distinct vocabulary (word types) found at least five times in the entire set of French documents in a trie. They considered two equivalence classes of characters, one being vowels, where any combination of vowels could be substituted for any other, the other being *k* sounds, where any combination of *c-k-qu* could substitute for any other. Thus *music* and *musique* would be considered equivalent. If it was possible to transform any of the English query words into any of the words in the trie of French words using either of the allowable substitutions, or by adding or deleting one letter, those French words were added to the query.

For our basic method, we assume that cognate words are vocabulary items which occur in two or more languages, such that they have both similar meanings and similar orthography. The degree to which two words, one Italian and its Spanish equivalent, are orthographically similar can be estimated using edit distance, which has for some time been used for automatic spelling correction (Wagner and Fischer, 1974). Character level alignment may be performed by the technique of dynamic programming. The difference between two word forms (called the edit distance) is taken to be the smallest number of operations required to transform one word into the other. The allowable operations include substitution (a single character in one word is replaced by a single character in the other), deletion of a single character from the first word, and insertion of a single character into the second word. For example, we can align the words *automobile* and *automóvil* as shown in figure 1.

Figure 1. Character-level alignment of *automobile* and *automóvil*

```
e   -> 0  deletion :
l   -> l  match :
i   -> i  match :
b   -> v  substitution :
o   -> ó  substitution :
m   -> m  match :
o   -> o  match :
t   -> t  match :
u   -> u  match :
a   -> a  match :
edit distance = 3
```

In order to transform the Italian *automobile* into the Spanish *automóvil*, three operations (two substitutions and a deletion) are required, and thus the edit distance is 3. The technique of dynamic programming finds the alignment which minimises the edit distance. To convert the edit distance into a measure of the closeness between two words, we use the following formula (McEnery & Oakes, 1996). This matching coefficient will be 1 for two identically spelled words, and 0 for two words with no characters at all in common.

$$matching\_coefficient = 1 - \frac{edit\_dis\tan ce}{length\_of\_longer\_word}$$

A vocabulary list of over 56,000 words was produced consisting of all the Spanish words occurring twice or more in a subset of the Spanish document set. Program clef5.c was written to align each word in the Italian query sets with each Spanish word in the lexicon in turn, using dynamic programming. A subjectively assigned threshold of 0.8 was chosen, such that any word in the Spanish lexicon with a matching coefficient of 0.8 or more with respect to the Italian query word was assumed to be a possible translation of that query word. This method is similar to that used by the Kantrowitz et al. stemmer (2000), an edit distance stemmer with complex edits corresponding to the most common suffixes. The first five Italian queries are shown in figure 2.

Figure 2. Five original Italian queries

```
*c141 lettera bomba per kiesbauer
*c142 christo impacchetta parlamento tedesco
*c143 conferenza pechino sulle donne
*c144 ribellioni sierra leone diamanti
*c145 importazioni giapponesi riso
```

Figure 3. Translation of Italian queries into Spanish queries

```
*c141
[bomba] [bomba] match = 1.000000
[bomba] [bombas] match = 0.833333
[bomba] [bombay] match = 0.833333
*c142
[christo] [christa] match = 0.857143
[christo] [christi] match = 0.857143
[christo] [cristo] match = 0.857143
[christo] [hristo] match = 0.857143
[parlamento] [apartamento] match = 0.818182
[parlamento] [palamento] match = 0.900000
[parlamento] [parlament] match = 0.900000
[parlamento] [parlamento] match = 1.000000
[parlamento] [parlamentos] match = 0.909091
*c143
[conferenza] [conferencia] match = 0.818182
[donne] [donner] match = 0.833333
*c144
[sierra] [cierra] match = 0.833333
[sierra] [pierra] match = 0.833333
[sierra] [serra] match = 0.833333
[sierra] [sierra] match = 1.000000
[sierra] [tierra] match = 0.833333
[leone] [leone] match = 1.000000
[leone] [leonel] match = 0.833333
[leone] [leones] match = 0.833333
[diamanti] [diamant] match = 0.875000
[diamanti] [diamante] match = 0.875000
*c145
[importazioni] [importacion] match = 0.833333
```

The Italian query sets were “translated” into Spanish using program clef5.c, as shown in figure 3. Query words of fewer than 4 characters were stoplisted. Query c141 shows three types of Spanish terms picked up in response to the Italian query word *bomba* (bomb). Firstly, we find an exact match with *bomba*, the correct Spanish equivalent. We also get an above threshold match with *bombas*, a grammatical variant (plural) of *bomba*. Finally we pick up the incorrect translation *Bombay*, a source of noise in our system. The overall Spanish query to be presented to the search engine is *bomba*, *bombas*, *Bombay*. No Spanish query terms were found corresponding to the Italian words *lettera* or *Kiesbauer.*

2. Second experimental run: cognates sought by regular sound changes

For the second experimental run, we used a more linguistically accurate definition of cognates, namely that cognate words are vocabulary items which occur in two or more historically related languages, such that they have similar meanings, and one can be transformed into the other by a predictable series of phonological changes. For example, Nothofer (1975) has manually produced tables showing the sound equivalences which occur in four languages spoken in or near the Indonesian island of Java, namely Javanese, Madurese, Malay and Sundanese, all of which originate from the common ancestor language Proto-Malayo-Javanic (PMJ). One example of such an equivalence is Javanese *d* = Madurese *jh* = Malay *j* = Sundanese *j*, as in the words for *road*, *dalan*, *jhalan*, *jalan* and *jalan* respectively. Such a system of sound correspondences was first described for Indo-European languages where it is referred to as Grimm’s Law, and it was later shown that such systems are found in all language families (Bloomfield, 1925). The task of identifying regular sound changes in bilingual word lists has been described by Guy (1994) as “Given a sample word list from two related languages, extract the probable rules for predicting any word of one language from that of the other”.

To find the regular sound changes found in a given language pair, the starting point is a bilingual word list where each word in one language is aligned with its translation. Single character substitutions can be identified using the Wagner & Fischer edit distance algorithm described in section 1. However, in their work on bilingual sentence alignment, Gale & Church (1993) introduced additional operations into the dynamic programming algorithm. While the original algorithm allows only for single character insertions, deletions and substitutions, Gale and Church also considered for example 2:1 correspondence, denoting that two sentences of one language correspond with just one of the other. They also allowed for the fact that some operations are more commonly encountered in real data that others, by assigning higher edit distances to less frequently encountered operations. In program Jakarta.c (Oakes, 2000) the allowed operations correspond to the list of the types of sound change which typically occur between related languages throughout the world given by Crowley (1992, Chapter 2). These include single character operations, operations with higher cardinality, operations which can only involve certain characters such as vowels, and operations which can only take place at certain positions (such as initial) in the two words. Program Jakarta.c was used to collate the sound changes discovered in a list of 63 word pairs taken from the introductions to Collins Gem Italian and Spanish dictionaries, examples of which are shown in figure 4 below:

Figure 4. Sample word pairs examined for regular sound changes

Abbreviazione	abreviatura
aggettivo	adjetivo
amministrazione	administración
avverbio	adverbio
agricoltura	agricultura
anatomia	anatomía
architectura	arquitectura
automobile	automóvil
biologia	biología
botanica	botánica

The sound changes found in the word pair *aggetivo* and *adjetivo* are shown in figure 5. These changes were a fusion of the double *t* in Italian to a single *t* in Spanish, and the dissimilation of the *gg* (producing a single sound) into *dj* (producing two separate sounds) in Spanish. All word pairs in the bilingual list were compared in this way, and the changes were collated to produce the list shown in figure 6, which includes the number of instances where a character remained as itself. *0* represents a null character. Pairs of words which require an above threshold edit distance of three to transform one into the other are deemed not to be cognate (such as *abbreviazione* and *abbreviatura*), and do not contribute to the tally of discovered sound changes.

Figure 5. Alignment of *aggettivo* and *adjetivo*

```
o   -> o  match :
v   -> v  match :
i   -> i  match :
tt  -> t  fusion :
e   -> e  match :
gg  -> dj dissimilation :
a   -> a  match :
cost = 2
```

Figure 6. Sound substitutions found in the bilingual word list

```
a   -> a   : 50
a   -> o   : 1
az  -> ac  : 1
b   -> b   : 6
c   -> c   : 18
d   -> d   : 1
e   -> é   : 1
e   -> 0   : 9
e   -> e   : 29
f   -> f   : 10
g   -> g   : 19
gg  -> dj  : 1
gg  -> uj  : 1
i   -> í   : 12
i   -> i   : 47
l   -> l   : 23
m   -> m   : 13
mm  -> m   : 1
n   -> n   : 23
o   -> ó   : 3
o   -> 0   : 1
o   -> o   : 35
o   -> u   : 4
p   -> p   : 6
q   -> q   : 1
r   -> r   : 26
s   -> s   : 8
ss  -> s   : 1
st  -> s   : 1
t   -> d   : 1
t   -> t   : 26
tt  -> ct  : 1
tt  -> t   : 1
u   -> ú   : 1
u   -> o   : 1
u   -> u   : 7
v   -> v   : 9
vv  -> dv  : 1
z   -> z   : 1
za  -> te  : 1
0   -> 0   : trivial
```

Based on the sound changes seen in figure 6, the basic edit distance program used for the first experimental run (clef5.c) was amended to form program sounds5.c which implements the following metric: a cost of 1 is assigned for each insertion, deletion or single character substitution not recognised as being regular, and a cost of

0 for each exact character match, deletion or single character substitution listed as being regular. The transformations regarded as being regular are shown in figure 7. As for the first experimental run, each Italian query term was matched against the Spanish lexicon, and all Spanish terms matching the Italian terms with an above threshold coefficient were included as Spanish query terms. A slightly higher threshold of 0.85 was used for the second experimental run, as the incorporation of the regular sound changes meant that true cognates matched with slightly higher coefficients, and raising the threshold would reduce the noise caused by false cognates being selected. The first five Spanish query sets produced for the second experimental run are shown in figure 8.

Figure 7. Sound changes used in the second experimental run

```
z -> c not initial
o -> u not initial
t -> c not initial
v -> d not initial
i -> í
o -> ó
u -> ú
g -> j not first or second character
delete terminal o
delete terminal e
```

Figure 8. Translation of Italian queries into Spanish queries using regular sound changes

```
*c141
[lettera] [lectura] match = 0.857143
[lettera] [lecturas] match = 0.875000
[bomba] [bomba] match = 1.000000
[bomba] [bombas] match = 1.000000
*c142
[christo] [chris] match = 0.857143
[christo] [christa] match = 0.857143
[christo] [christi] match = 0.857143
[christo] [cristo] match = 0.857143
[christo] [hristo] match = 0.857143
[parlamento] [palamento] match = 0.900000
[parlamento] [parlament] match = 1.000000
[parlamento] [parlamento] match = 1.000000
[parlamento] [parlamentos] match = 1.000000
*c143
[conferenza] [conferencia] match = 0.909091
[conferenza] [conferencias] match = 0.916667
[donne] [dunn] match = 1.000000
*c144
[sierra] [sierra] match = 1.000000
[sierra] [tierras] match = 0.857143
[leone] [leon] match = 1.000000
[leone] [león] match = 1.000000
[leone] [leone] match = 1.000000
[leone] [leones] match = 1.000000
[diamanti] [diamant] match = 0.875000
[diamanti] [diamante] match = 0.875000
[diamanti] [diamantes] match = 0.888889
*c145
[importazioni] [importacion] match = 0.916667
[importazioni] [importación] match = 0.916667
[importazioni] [importaciones] match = 0.923077
```

3. The search engine

For both experimental runs, the task was to translate the Italian query sets to Spanish query sets, then match the Spanish query sets against the Spanish document set using a search engine. Our search engine uses a very simple algorithm. The Spanish query sets are submitted in turn, and for each document in the Spanish document set, a score of 1 is given for each word in the document which matches a word in the query set. The overall score for each document is normalised by dividing it by the number of words in the document, and the documents are ranked, so that those with the best normalised matching score are presented first.

4. Conclusions

We believe that our results could be improved in future in a number of ways. Firstly a larger Spanish vocabulary could be produced using the entire Spanish document set. Secondly, we need to determine optimal matching coefficient thresholds which best discriminate between true and false word translations. Thirdly, we need a much larger bilingual word list to better determine the set of sound changes found in Italian and Spanish cognate words. Finally, program sounds5.c could be enhanced to allow regular multiple character substitutions.

We have demonstrated a method of generating target language query words using source language keywords and a list of regular sound changes, if the source and target languages are historically related, as they are in the case of Italian and Spanish, which share many cognate words. The differences with respect to Buckley et al.'s approach are firstly that linguistically motivated sound substitutions are used, and secondly that regular sound substitutions are used rather than just orthographic substitutions for homophones.

References

L Bloomfield, On the Sound System of Central Algonquian. Language 1, pp 130-156, 1925.

C Buckley, J Walz, M Mitra & C Cardi, “Using Clustering and Super Concepts Within SMART”, NIST Special Publication 500-240: The Sixth Text Retrieval Conference (TREC6), 2000. http://trec.nist.gov/pubs/trec6/t6_proceedings.html

T Crowley. An Introduction to Historical Linguistics. Oxford University Press, 1992.

W Gale & K Church. A Program for Aligning Sentences in Bilingual Corpora. Computational Linguistics, 19(1), pp 75-102, 1993.

J B M Guy. An Algorithm for Identifying Cognates in Bilingual Word Lists and its Applicability to Machine Translation. Journal of Quantitative Linguistics 1(1), pp 34-42, 1994.

M Kantrowicz, M Behrang & V Mittal, “Stemming and its Effects on TFIDF Ranking”, Proceedings of the 23rd ACM SIGIR Conference, Athens, Greece, 2000.

A M McEnery & M P Oakes ,“Sentence and Word Alignment in the CRATER Project”, in “Using Corpora for Language Research”, ed. J Thomas and M Short, Longman, pp 211-231, 1996.

B Nothofer. The Reconstruction of Proto-Malayo-Javanic. ‘s-Gravenhage:Martinus Nijhoff, 1975.

M P Oakes, Computer Estimation of Vocabulary in a Protolanguage from Word Lists in Four Daughter Languages, Journal of Quantitative Linguistics 7(3) 2000, pp 233-244.

R A Wagner & M J Fischer, The String to String Correction Problem. Journal of the ACM, 21, p 168, 1974.