Studying the Effect and Treatment of Misspelled Queries in Cross-Language Information Retrieval

Jesús Vilares[a,*] Miguel A. Alonso[a], Yerai Doval[a,b], Manuel Vilares[b]

[a] Grupo LYS, Departamento de Computación, Facultade de Informática, Universidade da Coruña
Campus de Elviña, 15071 – A Coruña (SPAIN)
{jesus.vilares, miguel.alonso, yerai.doval}@udc.es

[b] Grupo COLE, Departamento de Informática, E.S. de Enxeñaría Informática, Universidade de Vigo
Campus As Lagoas, 32004 – Ourense (SPAIN)
{yerai.doval, vilares}@uvigo.es

Abstract

In contrast with their monolingual counterparts, little attention has been paid to the effects that misspelled queries have on the performance of Cross-Language Information Retrieval (CLIR) systems. The present work makes a first attempt to fill this gap by extending our previous work on monolingual retrieval in order to study the impact that the progressive addition of misspellings to input queries has, this time, on the output of CLIR systems. Two approaches for dealing with this problem are analyzed in this paper. Firstly, the use of automatic spelling correction techniques for which, in turn, we consider two algorithms: the first one for the correction of isolated words and the second one for a correction based on the linguistic context of the misspelled word. The second approach to be studied is the use of character n-grams both as index terms and translation units, seeking to take advantage of their inherent robustness and language-independence. All these approaches have been tested on a from-Spanish-to-English CLIR system, that is, Spanish queries on English documents. Real, user-generated spelling errors have been used under a methodology that allows us to study the effectiveness of the different approaches to be tested and their behavior when confronted with different error rates. The results obtained show the great sensitiveness of classic word-based approaches to misspelled queries, although spelling correction techniques can mitigate such negative effects. On the other hand, the use of character n-grams provides great robustness against misspellings.

NOTICE: this is the authors' version of a work that was accepted for publication in Information Processing and Management. Changes resulting from the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. A definitive version was subsequently published in Information Processing and Management http://dx.doi.org/10.1016/j.ipm.2015.12.010

Keywords

Misspelled queries; Cross-Language Information Retrieval; Machine Translation; spelling correction, character *n*-grams.

* Contact author

1. INTRODUCTION

When facing the current context of globalization of the use of Internet and the Web, classic Information Retrieval (IR) systems (Manning et al., 2008) have to deal with the fact that in many cases the information available is written in a language different from that of its potential customers, the language they most probably use when submitting a query. In response to this issue, the field of Cross-Language Information Retrieval (CLIR) has arisen within the IR community.

In brief, CLIR is a particular case of IR where queries and documents are written in different languages (Nie, 2010; Kim et al., 2015; Peters et al., 2012; Grefenstette, 1998): given a query in one language (called *source language*), the system provides the user with the means and skills needed to find relevant documents written in another language (called *target language*). Most CLIR systems apply some kind of intermediate *translation stage* in order to convert the cross-language configuration to a classic monolingual configuration (i.e., with queries and documents written in the same language) that can be managed by classic IR systems. Within this framework, we refer to query-translation based CLIR, document-translation based CLIR and interlingua-based CLIR when queries, documents or both queries and documents are translated, respectively (Wu et al., 2008). Due to the fact that the translation of large document collections has serious practical limitations, works in this field have mostly focused on query translation (Nie, 2010).

In parallel, and also as a result of this phenomenon of globalization of access to information, it becomes increasingly necessary to have systems capable of operating on texts with misspelling errors, particularly in the case of queries (Guo et al., 2008). In fact, nowadays it is common to assume that some text-cleaning processing stage is needed in order to extract useful information from user-generated content, such as product reviews (Vilares et al., 2015a) or microblog entries (Vilares et al., 2015b; 2015c). In this work, we consider as *misspelling errors* those corresponding to typographical errors during writing, those due to the ignorance of the actual spelling of a word and those arising from the presence of noise in the generation process, e.g. OCR or speech recognition (Kukich, 1992). Since formal IR models were originally designed to work on texts without errors, their presence can substantially reduce system performance. To deal with this issue, another field has arisen within the IR community: *Tolerant Information Retrieval* (TIR) (Manning et al., 2008, Ch.3).

This article deals with the analysis of the impact of misspelled queries on CLIR systems and the design of Tolerant CLIR systems that are able to operate with such queries. Our practical experience suggests that the inability to deal with misspelled words is a major source of translation errors for the machine translation engines used in CLIR systems for query translation. In order to do this, we will take advantage of our previous experience both in the study of the impact of misspelled queries on monolingual IR (Vilares et al., 2011) and in character n-gram based CLIR (Vilares et al., 2016). Thus, we will make a comparative analysis of the effectiveness of two possible strategies which are reflected in three specific techniques with different levels of integration of knowledge and linguistic resources. These strategies are in line with the two generic state-of-the-art approaches to the problem of TIR: firstly, the use of words as working units and, secondly, the use of sub-words as working units. The proposed solutions have been subjected to different experiments in a from-Spanish-to-English retrieval context (i.e., queries submitted in Spanish over a collection of English texts). The methodology applied for this purpose allows us to test real human errors rather than artificially-generated ones, giving us a wide range of options.

To the best of our knowledge there are no similar works with this level of detail in a cross-language context. The work of Darwish and Magdy (2007), for example, although distantly-related to ours, differs significantly since it is focused on monolingual retrieval of scanned documents containing OCR errors, instead of multilingual retrieval with misspelling errors present in the queries, as is our case.

The structure of the rest of this paper is as follows. Section 2 describes our proposals for the treatment of queries with errors. Next, Section 3 discusses in detail our proposal based on the use of words as working units in conjunction with the use of spelling correction techniques, while Section 4 presents our proposal based on the use of character *n*-grams as working units both in the translation and retrieval stages. In Section 5, the methodology employed for testing is explained, and the experimental results obtained by means of it are analyzed in Section 6. Finally, Section 7 and Section 8 present, respectively, our conclusions and proposals for future work.

2. PROCESSING MISSPELLED QUERIES

Treatment of misspelled queries is usually based on replacing the original search algorithm for exact matches by a more flexible method allowing approximate ones. Having analyzed the state of the art, we consider here two different strategies for dealing with misspelled queries (Manning et al., 2008; Vilares et al., 2011): one that operates at word level and another one that operates at subword level.

As has been said before, the first of these strategies employs the word as working unit. This strategy relies on the use of dictionary-based Natural Language Processing (NLP) techniques in order to implement a query pre-processing stage for detecting and correcting the spelling mistakes that it may contain (Vilares et al., 2011). Once pre-processed, the query is translated and submitted to the system so the search process can be performed by a traditional IR engine. Our spelling correction solutions for this strategy will be described in Section 3.

At this point, we draw attention to the differences between IR and other areas of application of this type of automatic correction such as, for example, word processors. In this latter area, the usual solution consists of performing an ineffective first guess from which the system interacts with the user by showing several candidate corrections, asking the user to choose the right one. However, in the case of IR systems this type of approach is impractical, thus we require more complex, fully automatic error handling approaches with no further user intervention after entering the initial query.

On the other hand, the second strategy operates at sub-word level and consists of using character *n*-grams as processing units (Vilares et al., 2016). This kind of approach, to be described in greater detail in Section 4, can tackle the problem in a simpler way, independently of the degree of knowledge and linguistic resources available.

3. APPROACHES BASED ON SPELLING CORRECTION

As explained before, the first of the strategies referred to in this work involves pre-processing the original input query using NLP-based automatic correction techniques in order to detect and correct the spelling errors that it might contain. This strategy has been previously applied with success in monolingual IR (Vilares et al., 2011) and its extension to a cross-language domain is straightforward as no specific adaptation is required: the initial query is pre-processed and, once

corrected, processing continues as usual, it being first translated and then submitted to the retrieval engine.

Within this strategy based on spellchecking we can consider, in turn, two types of approach depending on the correction algorithm to be used:

1. *Correction of isolated words* (Savary, 2002): each word is corrected in isolation, without taking into account its surrounding words; thus we only have to consider those errors corresponding to out-of-vocabulary words. This technique may fail in detecting the so-called *real-word* errors, i.e. errors leading to different terms which do belong to the language. This is the case, for example, of the Spanish query *"apalabras* con errores"* ("you bespeak with errors") for *"palabras con errores"* ("words with errors") –where the asterisk denotes, from now on, a misspelled word. As can be seen, each of the terms of the misspelled query is correct when we consider them separately.
2. *Contextual correction* (Otero et al., 2007): by leveraging contextual linguistic information, techniques of this kind are able to detect errors such as the one shown above. Moreover, they allow the proposed corrections to be sorted and, this way, the most appropriate correction to be selected automatically given the context.

Both correction-based approaches are described more in depth in the following subsections, together with examples of their use.

3.1. Savary's Correction Algorithm

The first strategy consists of applying a global repair correction algorithm on isolated words (i.e. context is ignored) by means of finite-state automata, as proposed by Savary (2002). This algorithm is able to find all the words within a minimal edit distance with respect to the input misspelled word. The *edit distance* between two strings (Levenshtein, 1966) is defined as the number of *edit operations* to be applied to transform one string into the other. The edit operations to be considered are the following:

1. *Insertion* of a character, e.g. "*word*" → "*word<u>y</u>*".
2. *Deletion* of a character, e.g. "*wo<u>r</u>d*" → "*wod*".
3. *Substitution* of a character, e.g. "*wor<u>d</u>*" → "*wor<u>m</u>*".
4. *Transposition* of two adjacent characters, e.g. "*<u>wo</u>rd*" → "*<u>ow</u>rd*".

Savary's algorithm works as follows. The core of the spelling correction module is a *finite-state automaton* (FSA) $\mathcal{A} = (Q, \Sigma, \delta, q_0, Q_f)$ that recognizes the lexicon of the language, and where: Q is a set of states, Σ is a set of input symbols, δ: $Q \times \Sigma \rightarrow 2^Q$ is a function that defines the transition of the automaton, $q_0 \in Q$ is the initial state and $Q_f \subseteq Q$ is the set of final states.

The process starts as a standard recognizer, trying to go from the initial state to a final state by applying the transitions labeled with the characters of the input word. When the recognizer stops in a non-final state where there are no outgoing transitions labeled with the next character of the input word (i.e., it is not possible to consume the next character), it means that we have detected an error in

the input word. In response, we apply four classes of elementary *repair hypothesis* on the current configuration of the automaton in order to reach a new configuration in which the recognition process can continue. These hypotheses, each of them corresponding to one of the basic *edit operations* described above, have an associated predefined weight (for the experiments to be presented later in this paper, all operations were weighted equally) and work like this:

1. *Insertion*: ignores the current character of the input string and tries to continue from the current state.
2. *Deletion*: tries to follow every reachable state from the current one.
3. *Substitution*: ignores the current character of the input string and tries to follow every state accessible from the current state.
4. *Transposition*: only applicable when (i) the next two input characters are α and β, respectively; (ii) it is possible to reach an intermediate state q_j from the current state q_i through a transition labeled with β; and (iii) it is possible to reach a state q_k from the intermediate state q_j through a transition labeled with α. If these requirements are met, the algorithm tries to continue from the state q_k by ignoring the next two characters of the input string.

We must also take into account that there could be several consecutive errors and that an error may have been precipitated by an earlier erroneous repair. Therefore, these operations should be applied recursively until a correct configuration is reached, considering as starting points of the repair process the point where the error was detected and the previous configurations reached by the FSA. This algorithm reduces dynamically the search space by taking into account only the minimum number of repairs and by trying to reach the first repair as soon as possible.

Unfortunately, as a result of this correction process, the algorithm may return several candidate corrections that, from a morphological point of view, have a similar quality. This happens when several words exist simultaneously at minimal edit distance of the misspelled word.

Let us take, as an example, the sentence *"El balor* actua* de las cosaa*"* for *"El valor actual de las cosas"* ("The current value of [the] things"). For this input sequence, the algorithm would obtain the following possible candidate corrections for its misspelled words, all of them at minimal edit distance from the original misspelled term (one edit operation):

"balor*": *"calor"* ("heat"); *"valor"* ("value"); *"balar"* ("to bleat")
"actua*": *"actúa"* ("act [you]", "he acts"); *"actual"* ("current"); *"actuar"* ("to act")
"cosaa*": *"cosía"* ("I sewed", "he sewed"); *"costa"* ("coast"); *"cosan"* ("they sewed"); *"cosas"* ("things", "you sewed"); *cosa* ("thing", "I sewed", "he sewed")

leading to 3×3×5=45 word combinations, i.e. 45 possible solutions, as shown in Figure 1. The next question is which of them is the right one.

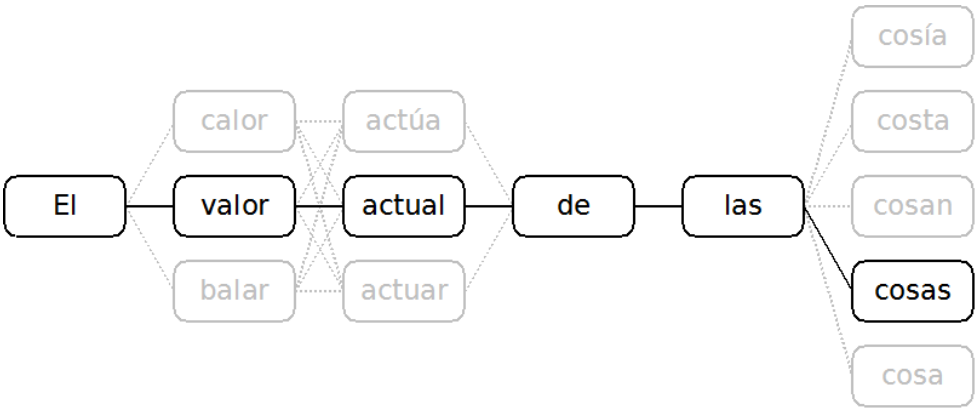

Figure 1. Compact representation of all the possible output combinations (total: 3×3×5=45 possibilities) obtained by means of Savary's algorithm for our sample sentence *"El balor* actua* de las cosaa*"* (the right sequence is shown highlighted).

3.2. Contextual Correction

3.2.1. Integrating Linguistic Knowledge

As we have seen in the previous section, Savary's original correction algorithm is not able to discriminate among different solutions being at the same minimal edit distance of the input term. However, we can discard most of them by making use of our own knowledge of the grammar of the language (Spanish in this case). For example, in the case of the last term (*"cosaa*"*), part of the candidates correspond to verb forms, and the rest of them correspond to noun forms (all of them feminine forms, either singular or plural). At the same time, that word comes after the sequence *"de las"* ("of the" [feminine plural]) and, according to Spanish grammar, what we should expect to find after that sequence is a plural feminine noun, not a singular feminine noun nor a verb form. So, since the only feminine plural noun among the candidate terms for *"cosaa*"* is *"cosas"*, we can keep it and dismiss the rest of the candidates for that term.

This is the same principle used by our second proposal for dealing with misspelled queries, an extension of Savary's original algorithm which we call *contextual spelling correction*. This algorithm uses the contextual linguistic information embedded within a POS-tagging process to prune the candidate corrections. Only those candidates that fit the morphosyntactic context of the term to be corrected will be kept (Otero et al., 2007).

A Part-Of-Speech (POS) tagging process consists of marking up a word in a text as corresponding to a POS, based on both its definition and its context (Manning and Schütze, 1999, Ch.10). According to our needs or availability, the POS tags to be used can be coarse-grained (e.g., when they only represent the grammatical category: noun, verb, adjective, determiner, etc.) or fine-grained (when they include additional morphosyntactic information such as gender, number, tense, etc.). In the case of Spanish, the use of fine-grained tags is convenient because of the need of reflecting the agreement in terms of gender and number between determiners, nouns and adjectives; the agreement in terms of number and person between subject and verb; etc.

3.2.2. Implementing the Approach

This proposal employs a stochastic morphosyntactic POS-tagger based on second order Hidden Markov Models (HMM) that makes use of the Viterbi algorithm to obtain, not only the correct sequence of POS tags corresponding to the input text (in this context, the most probable one), but also its associated probability (Manning and Schütze, 1999, Ch.10).

However, a new problem appears when trying to implement this proposal. Regular POS-taggers require a single form per input word, but this is not what we are working with since for a given position in the input sequence to be tagged, we may have more than one possible input form (one per candidate correction). One option to solve this problem consists on converting the original problem –a sequence of words where a given term may have more than one possible form–, a problem that cannot be managed by the tagger, into a different but equivalent problem that can be managed by it.

Thus, as described by Graña et al. (2002b), an alternative consists of expanding our initial input into the different sequences corresponding to each combination of candidate corrections, to process each one separately with the POS-tagger and, finally, to choose the most probable solution as the final output. However, this would have been a very inefficient process; in the case of our working

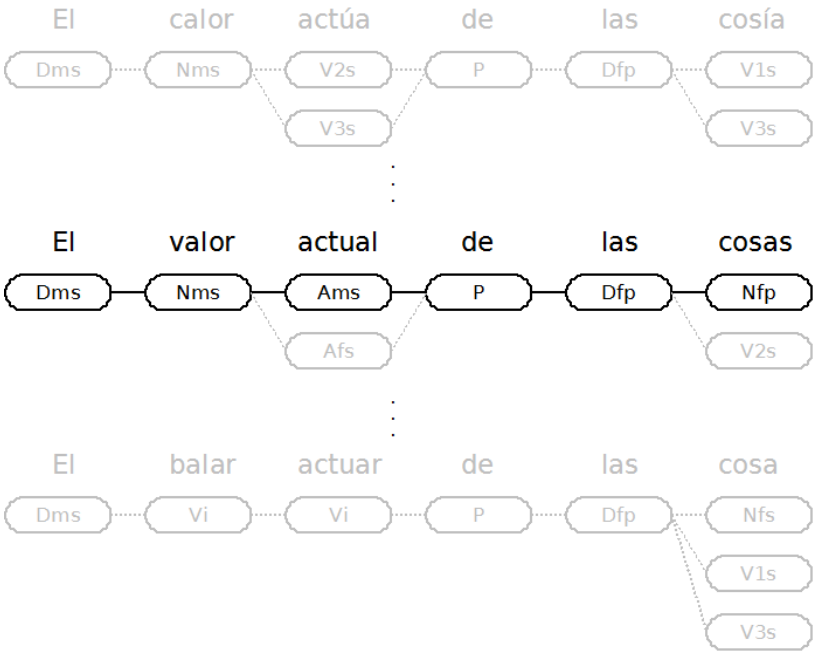

Figure 2. Input alternatives to be tagged corresponding to our sample sentence *"El balor* actua* de las cosaa*"* (the right sequence is shown highlighted). Tags are described in Appendix A.

sample, for example, we would have to tag 3×3×5=45 input sequences, one per possible candidate combination, as shown in Figure 2.

Looking for efficiency, our tested proposal does not integrates a regular HMM-based POS-tagger, but a more flexible variant designed by Graña et al. (2002a) that employs a dynamic extension of the Viterbi algorithm and that is applied on lattices instead of trellises. This allows us to represent a *word*::*POS-tag* pair in each arc, and then calculate the probability of each of the paths by means of an adaptation of the equations of the original Viterbi algorithm. As shown in Figure 3 for our working

sample, in a lattice words are denoted by the arcs rather than by the nodes, making it possible to represent a *candidate-correction*::*POS-tag* pair in each arc and then use Viterbi to obtain the most

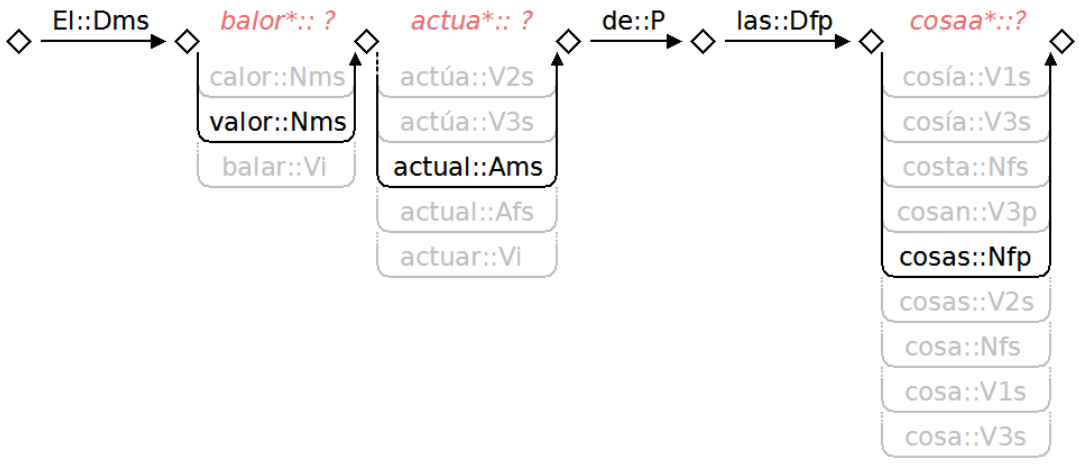

Figure 3. Alternative spelling corrections for our sample sentence *"El balor* actua* de las cosaa*"* represented in a lattice (the right sequence is shown highlighted). Tags are described in Appendix A.

likely sequence in a single pass. This allows us to discard those candidate corrections returned by Savary's original approach which do not correspond to the correct sequence of POS-tags.

4. APPROACHES BASED ON CHARACTER N-GRAMS

We define a *character n-gram* as a sequence of *n* characters within a word. Thus we can divide, for example, the word *"librero"* ("bookseller") into the following overlapping 4-grams (i.e. *n*-grams of length *n*=4): `-libr-`, `-ibre-`, `-brer-` and `-rero-`.

The advantages derived from the use of character *n*-grams for text processing —simplicity, efficiency, robustness, completeness and independence of the domain— have converted treatment at *n*-gram level into a standard technique of the state of the art (Robertson and Willett, 1998; Vilares et al., 2011). These advantages have not gone unnoticed by the IR research community. Classic IR systems usually employ knowledge and linguistic resources such as lists of stopwords, stemmers, lexicons, thesauri, taggers and so on. However, tokenization in *n*-grams does not require any of these: queries and documents are merely tokenized in overlapping *n*-grams, being then processed by the retrieval engine like any other term. In this way, *n*-gram tokenization constitutes a language- and domain-independent approach (Leveling and Jones, 2010). Moreover, as shown by McNamee and Mayfield (2004b; 2004a) and Robertson and Willett (1998), the employment of *n*-grams matching is itself an inherent mechanism of standardization of terms that can work with a variety of languages without any additional processing.

4.1. N-Gram Based CLIR

The use of character *n*-grams as working unit in regular monolingual IR systems is quite simple and does not raise any particular problem, thus allowing us to make use of their advantages. However, in the case of Cross-Language IR those advantages are limited by the intermediate

translation stage, which can only be performed at word or phrase level, whereby the query must be tokenized in *n*-grams *after* being translated.

Moreover, the translation process itself is also very sensitive to misspellings, unfamiliar words, the lack of appropriate linguistic resources, etc. This means, for example, that a misspelled word in the original query could not be translated correctly, so that further post-processing with *n*-grams loses much of its ability to find approximate matches. Taking Google Translate as example, the input misspelled word *"librp*"*, for *"libro"* ("book"), remains untranslated as *"librp"*. As we have stated before, our practical experience suggest that the inability to deal with misspelled words is a major source of translation errors for machine translation engines.

However, if we could find a way to eliminate or, at least, to reduce such restrictions when translating, so that complete words are not required, CLIR systems could also benefit from the advantages that the use of character *n*-grams as processing units can provide, not only as indexing units, but also as translation units. McNamee and Mayfield (2004b) were pioneers in this field, employing an algorithm for *n*-grams translation based on the alignment of a parallel corpus at the *n*-gram level by using statistical techniques. Unfortunately, this first approach was slow and rigid, and thus of limited interest. Subsequently Vilares et al. (2016) presented an alternative system based on a different process for generating alignments for *n*-grams, preserving the benefits of the system by McNamee and Mayfield (2004b) but eliminating its main disadvantages. At this point, we should emphasize the fact that, although valid for retrieval purposes, this *n*-gram level translation is not a proper translation from a linguistic point of view, so we should rather consider it as a kind of *pseudo-translation*; however, for simplicity, we will keep using the term *translation* when referring to it.

In any case, this kind of approach allows us, in the context of CLIR, to extend the inherent benefits of using *n*-grams as processing unit to the translation process. This way we can avoid some of the limitations of the dictionary-based techniques, such as the need to normalize the words or the inability to translate unknown terms. Moreover, as this solution does not use any particular language-dependent processing, it can be applied when the availability of language resources is scarce, which, as stated by Rehm and Uszkoreit (2011) and contrary to what one might think, is not unusual, even for major European languages.

4.2. Translating n-Grams

Our *n*-gram level translation system (Vilares et al., 2016) is based, in turn, on an alignment algorithm that works at character *n*-gram level, taking as input a parallel corpus in the languages to work with. This alignment is performed progressively in two phases: from a parallel corpus to a word-level alignment and next, from a word-level to a character *n*-gram level alignment. Next, we describe each phase in more detail.

Firstly, the input parallel corpus in the required source and target languages is aligned at word level using the well-known statistical tool GIZA++ (Och and Ney, 2003), obtaining as output the translation probabilities between words in both languages. Such alignments are bidirectional (Koehn et al., 2003), that is, we only accept an alignment *source-language*→*target-language* (w_s, w_t) –where w_s and w_t denote the word in the source and target language, respectively–, if and only if there also exists the corresponding alignment *target-language*→*source-language* (w_t, w_s). Moreover, those alignments below a probability threshold W=0.15 are discarded. Thus, this first step plays the role of a filter, since we will consider for further processing only those *n*-grams alignments corresponding to aligned words.

The character *n*-gram level alignment, properly speaking, is performed during the second phase of the process. For this purpose we compute statistical association measures (Evert, 2005, Ch.3) based on co-occurrences of the character *n*-grams belonging to words aligned in the previous phase. Thus, given the *n*-gram pair (g_s, g_t) –where g_s denotes the character *n*-gram in the source language and g_t its candidate *n*-gram translation–, their co-occurrence frequencies can be organized in a *contingency table*. This table results from classifying the co-occurrences between the *n*-grams of the input *n*-gram pair and other *n*-grams present in pairs of aligned words, like this:

	$g_t \in w_t$	$g_t \notin w_t$	
$g_s \in w_s$	O_{11}	O_{12}	$= R_1$
$g_s \notin w_s$	O_{21}	O_{22}	$= R_2$
	$= C_1$	$= C_2$	$= N$

(1)

where the first row corresponds to observations where a word in the source language (w_s) contains the *n*-gram g_s, and the second row to those in which such word does not. The same occurs for the columns in the case of the word in the target language (w_t) and g_t. The figures of these cells are called *observed frequencies*: O_{11}, for example, corresponds to the number of alignments where the source word

w_s	w_t	probability
libro	*book*	0.833
librero	*bookseller*	0.454
librería	*bookshop*	0.202
librería	*bookstore*	0.537

w_s	w_t	probability
libro	*gift*	0.005
librero	*advice*	0.001
librería	*address*	0.003
librería	*open*	0.001

Table 1. Example Spanish-English word-level alignments and their associated probabilities.

contains g_s and its translation candidate contains g_t, while O_{12} is the number of alignments where the source word contains g_s but the candidate translation does not contain g_t, etc. R_1 and R_2 are the partial sums for each row of these frequencies, and C_1 and C_2 are the sums per column. The total number of pairs considered, N, is the sum of all observed frequencies. However, when making this calculations, we must weight the frequency of observations based on the probability associated with these alignments. This is due to the fact that they are not actual, but only probable, observations of co-occurrences since GIZA++ uses a statistical model for the initial word-level alignment, a model which computes the translation probability for each pair of co-occurrent words (Och and Ney, 2003). Therefore, the same source word may be aligned with more than one candidate translations at the same time, each one having a different probability. Let us consider as an example the case of the Spanish words *"libro"* ("book"), *"librero"* ("bookseller") and *"librería"* ("bookshop", "bookstore [US]") for which a possible Spanish-English word-level alignment, with its corresponding probabilities, is shown in Table 1.

In this case, the observed frequency O_{11} corresponding to the character *n*-gram pair (`-libr-`, `-book-`) would not be O_{11}=4 as we could suppose at first glance, but O_{11}=2.026. This is due to the fact that, although that *n*-gram pair co-occurs in four word-level alignments: (*"libro"*, *"book"*), (*"librero"*, *"bookseller"*), (*"librería"*, *"bookshop"*) and (*"librería"*, *"bookstore"*), such co-occurrences should be weighted according to the probability of their alignments:

O_{11}(-libr-,-book-) = 0.833 for (*"libro"*, *"book"*) + 0.454 for (*"librero"*, *"bookseller"*) + (2)

0.202 for (“*librería*”, “*bookshop*”) + 0.537 for (“*librería*”, “*bookstore*”) = 2.026

Once we have generated the contingency table corresponding to the pair of *n*-grams under consideration, we compute its association measure, obtaining the best results by using the *Dice's similarity coefficient* and the *log-likelihood* (Vilares et al., 2016), defined according to (Evert, 2005, Ch.3):

$$Dice = \frac{2O_{11}}{R_1+C_1} \qquad log\text{-}likelihood = 2\sum_{i,j} O_{ij}\log\frac{NO_{ij}}{R_iC_j} \tag{3}$$

5. DESIGNING THE EXPERIMENTS

5.1. Evaluation Framework

We have chosen for our experiments a *from-Spanish-to-English* CLIR set-up (queries in Spanish with English documents) for the following reasons:

1. The inclusion of English was convenient, it being the dominant language on the Web. As more information on the Web is available in English than in any other language, English should play the role of target language.
2. Many users, even if they understand English, have trouble talking it and writing in it, so they prefer to use their mother tongue as source language.
3. Given the wide variety of morphological processes present in Spanish, the treatment of spelling errors in this language is a real challenge (Vilares et al., 2004).

Regarding the evaluation corpus, the document collection to be used is the so-called *LA Times 94* (56,472 documents, 154 MB), previously employed in the *robust task* of the *ad hoc track* of CLEF 2006 (Di Nunzio et al., 2006), which reused queries from previous CLEF Initiative (2015) events. The other English sub-collection, the so-called *Glasgow Herald 95*, could not be used because having been introduced later than the *LA Times 94*, it does not provide relevance references (the so-called *qrel* files) for most queries. With respect to queries, we have employed the *training set* provided for such task (60 *topics* that are available, among other languages, in Spanish and English): topics C050-C059, C070-C079, C100-C109, C120-C129, C150-159 and C180-189. Each of them consists of three fields: *title*, a short title as the name implies; *description*, a brief phrase description; and *narrative*, a short text specifying its criteria of relevance. In order to be able to analyze in greater detail the influence which both query length and redundancy of the information contained in it have on system performance, we have considered two series of experiments regarding the topic fields used to generate the query:

1. *short queries*: using only the *title* field (average length: 3 words);
2. *mid-length queries*: using the fields *title* and *description*, the mandatory configuration required in CLEF competitions, used to obtain the scores for the official ranking (average length: 10 words).

These types correspond to the length and complexity of queries usually employed in commercial search engines and other IR systems (Bendersky and Croft, 2009; Jansen et al., 2000).

With respect to other resources used, both correction-based approaches require a dictionary of the language and, in the case of contextual correctness, a training corpus for POS-tagging is also needed. In both cases we have used the Spanish corpus of MULTEXT-JOC (Véronis, 1999), developed within the MULTEXT project (http://cordis.europa.eu/result/rcn/21271_en.html) and containing approximately 200,000 words with POS-tags attached, belonging to a lexicon of more than 15,000 words.

5.2. Generating the Misspelled Queries

As in the case of our previous experiments in monolingual IR (Vilares et al., 2011), to evaluate the different approaches we will introduce spelling errors into the input topics in order to analyze their impact on the results. We have tested a wide range of *error rates* T:

$$T \in \{0\%, 10\%, 20\%, 30\%, \ldots, 60\%\} \tag{4}$$

where a rate value T implies that $T\%$ of the words of the topic contain errors, with T=0% corresponding to the *original topic* (i.e. no extra errors added) and, as we shall see, with the limit of T=60% being imposed by the test set. This wide range of values allows us to study the behavior of the approaches even for the high error rates that characterize noisy environments like those in which the input is obtained from mobile devices based on handtyping or handwriting (e.g. smart-phones, digital pens or graphic tablets) or obtained from speech recognition systems like those integrated within the so-called *personal assistants* for mobile devices –e.g., SIRI (http://www.apple.com/ios/siri/). It should be noted that the use of such high rates is by no means excessive: for short queries, as is the case of our experiments, the error rate must necessarily be high in order for it to be reflected in the queries. For example, in the case of a two-word query, an error rate of 50% would be required in order to have one error per word on average; for three-word queries, it would require an error rate of 33% and so on.

Unlike our previous monolingual experiments (Vilares et al., 2011), this time we have not worked with artificially generated errors but with real human errors only. These errors are more complex to generate and control, but they are also much closer to the reality than artificial errors, and therefore more interesting to study. Thus, we have re-used those topics containing human errors that were already employed in those monolingual experiments. Such topics had been generated like this. Firstly, with the help of third parties not involved in this research, a *pool* of handmade copies of the original topics was created. For this purpose, our collaborators (a group of eight people, consisting of PhD students and lecturers) were asked to type at least three copies of the original topics each. They were instructed to make them by typing fast or in noisy environments –e.g., while watching TV–, and not to correct any error they might make when typing. Next, the 27 copies obtained were aligned and, for each word, the most common mistake made was identified. This allowed us to identify copy errors for 65.62% of the terms (i.e., T=65.62%, 60% in practice). Finally, from this information, test sets were generated by progressively increasing the error rate so that it was cumulative in order to avoid distortions —that is, if a given error appears for T=20%, it should remain there for T>20%. More details about the generation of the input query set are available in (Vilares et al., 2011).

5.3. Indexing and Retrieval

Our CLIR system integrates the open source TERRIER IR platform (Ounis et al., 2007), configured for using a DFR Inverse Document Frequency ranking model with Laplace after-effect and normalization 2 (denoted as `InL`). Regarding indexing and retrieval processes, input documents and

queries are processed in a different way depending on the processing unit employed: words or character *n*-grams.

In the case of word-based approaches, text (either queries or documents) is normalized using a classic *stemming*-based approach. For this purpose, our system employs the Snowball *stemmer* (http://snowball.tartarus.org), based on Porter's algorithm, and the list of *stopwords* provided by the University of Neuchâtel (http://www.unine.ch/info/clef/). Both resources are well known and widely used. This being a CLIR process, the query was translated using Google Translate (http://translate.google.es) before being normalized. We consider, in turn, three cases:

- *stemming* (our baseline): the query is translated as is, with errors
- *Savary*: the query is corrected before submission by applying the Savary's correction algorithm on isolated words described in Section 3.1.
- *contextual*: this time the query is corrected by using our contextual correction algorithm, described in Section 3.2.

Furthermore, in the case of our *n*-gram based approach (denoted *4-grams*), the text is normalized by lowercasing it and removing punctuation before being *tokenized* into *n*-grams (McNamee and Mayfield, 2004b) to be then indexed (in the case of documents) or translated (in the case of queries). Although there are many possible configurations, we have chosen to use that one for which we had attained the best results in (Vilares et al., 2016). Thus, 4-grams (i.e. *n*-grams of 4 characters) were used, the computation of *n*-grams alignments was performed using *log-likelihood* as association measure, and the translation of the input query was performed using a so-called *range selection algorithm*. In this technique, each *n*-gram of the original query is replaced by its *H n*-gram candidate translations with the highest association measures, in this case using H=1. Finally, the translated query is sent to the retrieval engine.

Note that, intentionally, we have not employed techniques for query expansion nor relevance feedback in order to study the behavior of the approaches without introducing distortions in the results derived from the integration of other techniques (Vilares et al., 2011; 2016).

6. EXPERIMENTAL RESULTS

Since this study is focused on the negative effects that the presence of misspelled query terms has on CLIR systems performance, the following metrics were taken as reference for our analysis:

1. The loss, in percentage, of the Mean Average Precision (MAP) obtained for those topics containing a given rate T of misspelled words, with respect to MAP previously obtained for the original topics (i.e. for T=0%).
2. The resulting increase in the number of queries for which relevant documents have not been retrieved when comparing the results obtained for those topics with a given rate T of misspelled words, with those ones previously obtained for the original topics.

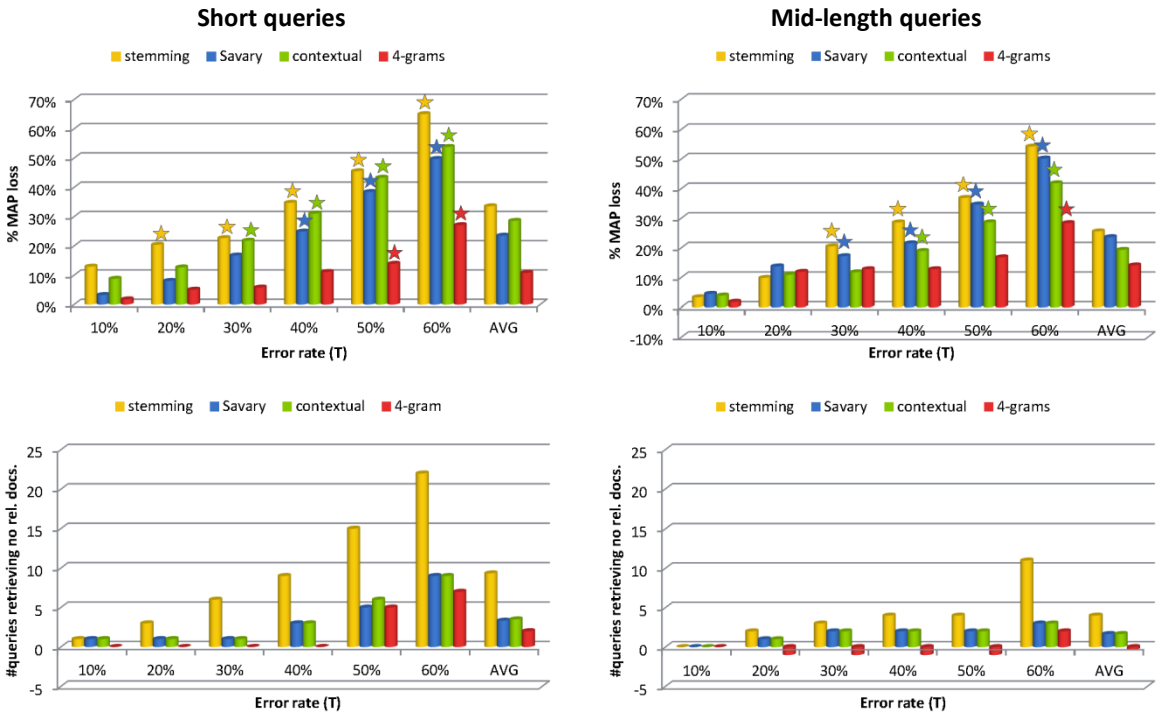

Figure 4. Charts showing the performance drop obtained in our experiments for short (left) and mid-length queries (right) with respect to the original queries (i.e. *T*=0%, no extra errors added) in terms of both the loss in percentage of the MAP obtained (top) and the number of queries for which relevant documents have not been retrieved (bottom). Note that shorter bars mean better results.

The figures obtained for each error rate *T* are presented as charts in Figure 4. For a more coarse-grained view of these results, the macro-averaged mean obtained for a given metric for each approach is also shown on the last bar group of each chart (*AVG*). Moreover, in the case of MAP loss, a star (★) is shown on top of the corresponding bar when that loss is statistically significant; for this purpose, *two-tailed t-tests* over MAP values with α=0.05 have been used.

Regarding the use of words as processing unit, our first results correspond to our stemming-based *baseline* (*stemming* runs, represented as yellow bars in the charts). As can be seen in Figure 4, these results show a significant negative impact of errors on the behavior of the system with respect to both MAP loss and the increase in the number of unanswered queries, even for the lowest error rates, and these results are even worse in the case of short queries. As shown in their corresponding *AVG* bar groups, we have obtained a macro-averaged mean drop of 34% (significant for T≥20%) for short queries compared to 26% (significant for T≥30%) for mid-length queries. The reason for this behavior is that the shorter the query, the greater the relative importance of each term, since the limited context available in those cases does not compensate for the loss of the information provided by each missing term.

On the other hand, results show that the use of correction-based techniques has a clearly positive effect that reduces this loss, although the behavior varies depending on the technique used and the type of query. In the case of using Savary's algorithm for the correction of isolated words (*Savary* runs, shown as blue bars), it can be seen in the corresponding *AVG* bar groups that, in absolute terms, its behavior is remarkably stable with respect to MAP loss, obtaining very similar results: a drop of 24% regardless of the length of the query. The case of contextual correction (*contextual* runs, shown

as green bars) is different, as it behaves much better with longer queries because, in principle, the linguistic context of the shorter queries is much more restricted, limiting its applicability. Thus, as can be seen in the corresponding *AVG* bar group, Savary's algorithm performs better than contextual correction in the case of short queries: a 24% drop (significant only for T≥40%) for Savary's vs. 29% (still significant for T≥30%) for contextual correction. In the case of longer queries, on the contrary, it is the contextual algorithm which behaves considerable better than Savary's algorithm: a 24% drop (significant for T≥30%) for Savary's vs. 19% (significant for T≥40%) for contextual correction.

Finally, in the case of using character *n*-grams instead of words as working unit (*4-grams* runs, shown as red bars), our results confirm their inherent robustness, even in a multilingual context like this. In these results, *n*-grams show to have a loss of accuracy and a number of unanswered queries clearly lower than those of our word-based approaches, particularly for very high error rates and also in the case of short queries. *n*-Grams show a much more robust behavior than our word-based baseline (*stemming*, yellow bars), as can be seen in the corresponding *AVG* bar groups: a drop of 11% (significant only for such a high rate as T≥50%) for *n*-grams vs. 34% (still significant for T≥20%) for the baseline, in the case of short queries; and a 14% drop (significant for T≥60%) for *n*-grams vs. 26% (significant for T≥30%) for the baseline, in the case of mid-length queries. Furthermore, *n*-grams are also able to clearly outperform correction-based approaches: even taking the best corrections, we obtain a drop of 11% (significant for T≥50%) for *n*-grams vs. 24% (significant for T≥40%) for Savary's algorithm in the case of short queries; and 14% (significant only for T≥60%) for *n*-grams vs. 19% (significant for T≥40%) for contextual correction in the case of mid-length queries. Moreover, *n*-grams achieve this high performance without needing to apply any specific processing for error handling.

7. CONCLUSIONS AND FUTURE WORK

Throughout the present work, we have analyzed the harmful effects of misspellings in queries in Cross-Language Information Retrieval environments, taking *a from-Spanish-to-English* configuration (queries made in Spanish on a collection in English) as a case study. We have considered several strategies and approaches to address this problem, a must-have step for developing more robust multi-language information retrieval systems.

The first strategy we have studied is a classic one based on the use of words as indexing and translation units. In this case we consider the use of spellchecking mechanisms for the treatment of errors before the query translation phase, presenting two alternatives for this purpose. Firstly, the use of Savary's global correction algorithm, which processes each word in isolation, returning the candidate corrections within a minimal editing distance, probably introducing noise when several alternative corrections exist and all of them are taken into consideration. Secondly, the use of a contextual correction algorithm that allows us to filter those candidate corrections based on their morphosyntactic context, returning only those corrections that agree with it.

In the case of our second strategy, we consider the use of character *n*-grams instead of words as processing unit, both for indexing and translation. This allows us to work directly with the original query with errors, since the alignments are made at subword level and thus we are able to establish partial correspondences with those parts of the word which are free of errors.

The results of these experiments are consistent with those previously obtained in the case of monolingual IR (Vilares et al., 2011). Firstly, our CLIR tests have shown again that word-based

approaches are highly sensitive to the presence of errors in the query, particularly for short queries, although the use of correction mechanisms can significantly reduce their negative effects. Our results also suggest that Savary's algorithm is more appropriate in the case of shorter queries, while the contextual correction algorithm shows higher performance for longer queries. Secondly, our strategy based on character *n*-grams has shown great strength too, with a drop in performance significantly lower than that attained when correction mechanisms were applied. Moreover it should also be pointed out that this *n*-gram based approach is a *light* one from the point of view of the knowledge resources employed, because it is not based on any particular language-dependent processing, so it can be used for a wide variety of languages, even when the availability of linguistic information and resources is reduced. Other more traditional CLIR approaches require language specific resources such as lists of stopwords, dictionaries, stemmers, POS-taggers, a training corpus and so on, which, contrary to what it may appear, are not always available, even for major European languages, as shown by Rehm and Uszkoreit (2011).

Regarding future work, we intend to work mainly on improving the translation process of character *n*-grams in order to increase its quality for retrieval applications. Moreover, from a pragmatic point of view, and following the example of the research community, we intend to study the application of our character *n*-gram based approach to our current research lines in microblog text processing for text normalization (Pennell and Liu, 2014), sentiment analysis (Aisopos et al., 2012) and language identification tasks (Lui and Baldwin, 2014). At this respect, it should be noted that Twitter and other microblogging services are very noisy multilingual environments, for which specialized linguistic resources are still very scarce, particularly for non-English languages. As explained before, character n-gram based processing is specially accurate for its application in this kind of contexts.

Acknowledgements

This research has been partially funded by the Spanish Ministry of Economy and Competitiveness (MINECO), through projects FFI2014-51978-C2-1-R and FFI2014-51978-C2-2-R; and by the Autonomous Government of Galicia, through the Galician Network for Language Processing and Information Retrieval–RedPLIR (grant CN2014/034). Moreover, Yerai Doval is funded by the Spanish State Secretariat for Research, Development and Innovation (which belongs to MINECO) and by the European Social Fund (ESF) under a FPI fellowship (ref. BES-2015-073768) associated to project FFI2014-51978-C2-1-R.

References

Aisopos, F., Papadakis, G., Tserpes, K. and Varvarigou, T., 2012. Content vs. context for sentiment analysis: A comparative analysis over microblogs. In: Proceedings of the 23rd ACM Conference on Hypertext and Social Media (HT'12). New York: ACM, pp. 187-196. DOI: 10.1145/2309996.2310028.

Bendersky, M. and Croft, W.B., 2009. Analysis of long queries in a large scale search log. In: Proceedings of the 2009 Workshop on Web Search Click Data (WSCD'09). New York: ACM, pp. 8-14. DOI: 10.1145/1507509.1507511.

Carmel, D., Cohen, D., Fagin, R., Farchi, E., Herscovici, M., Maarek, Y.S. and Soffer, A., 2001. Static index pruning for information retrieval systems, In: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (ACM SIGIR'01). New York: ACM, pp. 43-50.

CLEF Initiative, 2015. Conference and Labs of the Evaluation Forum (formerly known as Cross-Language Evaluation Forum), http://www.clef-initiative.eu (visited on December 2015).

Darwish, K. and Magdy, W., 2007. Error correction vs. query garbling for Arabic OCR document retrieval. ACM Transactions on Information Systems (ACM TOIS); 26 (1): 5.

Evert, S., 2005. The statistics of word cooccurrences: word pairs and collocations. PhD. thesis, Universität Stuttgart. Available at http://elib.uni-stuttgart.de/opus/volltexte/2005/2371 (visited on December 2015).

Graña, J., Alonso, M.A. and Vilares, M., 2002a. A common solution for tokenization and part-of-speech tagging: One-pass Viterbi algorithm vs. iterative approaches. In: Text, Speech and Dialogue, Vol. 2448 of Lecture Notes in Computer Science. Berlin-Heidelberg-New York: Springer-Verlag, pp. 3-10.

Graña, J., Barcala, F.M. and Vilares, J., 2002b. Formal methods of tokenization for part-of-speech tagging. In: Computational Linguistics and Intelligent Text Processing, Vol. 2276 of Lecture Notes in Computer Science. Berlin-Heidelberg-New York: Springer-Verlag, pp. 240-249.

Grefenstette, G. (ed), 1998. Cross-Language Information Retrieval. Vol. 2 of The Kluwer International Series on Information Retrieval. Boston: Kluwer Academic Publishers.

Guo, J., Xu, G., Li, H. and Cheng, X., 2008. A unified and discriminative model for query refinement. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (ACM SIGIR'08). New York: ACM, pp. 379-386.

Jansen, B.J., Spink, A. and Saracevic, T., 2000. Real life, real users, and real needs: A study and analysis of user queries on the web. Information Processing & Management; 36 (2): 207-227.

Kim, S., Ko, Y. and Oard, D.W., 2015. Combining lexical and statistical translation evidence for cross-language information retrieval. Journal of the Association for Information Science and Technology (JASIST); 66 (1): 23-39.

Koehn, P., Och, F.J. and Marcu, D., 2003. Statistical phrase-based translation. In: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (NAACL'03). Morristown (NJ): ACL, pp. 48-54.

Kukich, K., 1992. Techniques for automatically correcting words in text. ACM Computing Surveys (CSUR); 24 (4): 377-439.

Leveling, J. and Jones, G.J.F., 2010. Sub-word indexing and blind relevance feedback for English, Bengali, Hindi, and Marathi IR. ACM Transactions on Asian Language Information Processing (ACM TALIP); 9 (3): 12.

Levenshtein, V.I., 1996. Binary codes capable of correcting deletions, insertions and reversals. Soviet Physics Doklady; 10 (8): 707-710.

Lui, M. and Baldwin, T., 2014. Accurate language identification of Twitter messages. In: Proceedings of the 5th Workshop on Language Analysis for Social Media (LASM 2014). Stroudsburg (PA): ACL, pp. 17-25.

Manning, C.D., Raghavan, P. and Schütze, H., 2008. Introduction to Information Retrieval. New York: Cambridge University Press.

Manning, C.D. and Schütze, H., 1999. Foundations of Statistical Natural Language Processing. Cambridge (MA): The MIT Press.

McNamee, P. and Mayfield, J., 2004a. Character N-gram Tokenization for European Language Text Retrieval. Information Retrieval; 7 (1-2): 73-97.

McNamee, P. and Mayfield, J., 2004b. JHU/APL experiments in tokenization and non-word translation. In: Comparative Evaluation of Multilingual Information Access Systems, Vol. 3237 of Lecture Notes in Computer Science. Berlin-Heidelberg-New York: Springer-Verlag, pp. 85-97.

Nie, J.Y., 2010. Cross-Language Information Retrieval. Vol. 8 of Synthesis Lectures on Human Language Technologies. San Rafael (CA): Morgan & Claypool Publishers.

Di Nunzio, G.M., Ferro, N., Mandl, T. and Peters, C., 2006. CLEF 2006: Ad Hoc Track Overview. In: Results of the CLEF 2006 Cross-Language System Evaluation Campaign, Working Notes of the CLEF 2006 Workshop; pp. 21-34. Available at (CLEF Initiative, 2015).

Och, F.J. and Ney, H., 2003. A systematic comparison of various statistical alignment models. Computational Linguistics; 29 (1): 19-51. GIZA++ toolkit available at https://github.com/moses-smt/giza-pp (visited on December 2015).

Otero, J., Graña, J. and Vilares, M., 2007. Contextual Spelling Correction. In: Computer Aided Systems Theory, Vol. 4739 of Lecture Notes in Computer Science. Berlin-Heidelberg-New York: Springer-Verlag, pp. 290-296.

Ounis, I., Lioma, C., Macdonald, C. and Plachouras, V., 2007. Research directions in Terrier: A search engine for advanced retrieval on the Web. Novática/UPGRADE Special Issue on Web Information Access; 8 (1): 49-56. Terrier toolkit available at http://www.terrier.org (visited on December 2015).

Pennell, D.L. and Liu, Y., 2014. Normalization of informal text. Computer Speech and Language; 28 (1): 256-277.

Peters, C., Braschler, M. and Clough, P., 2012. Multilingual Information Retrieval: From Research to Practice. Berlin-Heidelberg-New York: Springer-Verlag.

Rehm, G. and Uszkoreit, H. (eds), 2011. META-NET White Paper Series. Berlin-Heidelberg-New York: Springer. Available at http://www.meta-net.eu/whitepapers (visited on December 2015).

Robertson, A.M. and Willett, P., 1998. Applications of n-grams in textual information systems. Journal of Documentation; 54 (1): 48-69.

Savary, A., 2002. Typographical nearest-neighbour search in a finite-state lexicon and its application to spelling correction. In: Implementation and Application of Automata, Vol. 2494 of Lecture Notes in Computer Science. Berlin-Heidelberg-New York: Springer-Verlag, pp. 251-260.

Véronis, J., 1999. MULTEXT-Corpora. An annotated corpus for five European languages. CD-ROM. ELRA/ELDA.

Vilares, D., Alonso, M.A. and Gómez-Rodríguez, C., 2015a. A syntactic approach for opinion mining on Spanish reviews. Natural Language Engineering; 21 (1): 139-163. DOI: 10.1017/S1351324913000181.

Vilares, D., Alonso, M.A. and Gómez-Rodríguez, C., 2015b. A linguistic approach for determining the topics of Spanish Twitter messages. Journal of Information Science (JIS); 41(2): 127-145. DOI: 10.1177/0165551514561652.

Vilares, D., Alonso, M.A. and Gómez-Rodríguez, C., 2015c. On the usefulness of lexical and syntactic processing in polarity classification of Twitter messages. Journal of the Association for Information Science and Technology (JASIST); 66 (9):1799-1816. DOI: 10.1002/asi.23284.

Vilares, M., Otero, J. and Graña, J., 2004. On asymptotic finite-state error repair. In: String Processing and Information Retrieval, Vol. 3246 of Lecture Notes in Computer Science. Berlin-Heidelberg-New York: Springer-Verlag, pp. 271-272.

Vilares, J., Vilares, M., Alonso, M.A. and Oakes, M.P., 2016. On the Feasibility of Character n-Grams Pseudo-Translation for Cross-Language Information Retrieval Tasks. Computer Speech and Language; 36: 136-164. DOI: 10.1016/j.csl.2015.09.004.

Vilares, J., Vilares, M. and Otero, J., 2011. Managing Misspelled Queries in IR Applications. Information Processing & Management; 47 (2): 263- 286. DOI:10.1016/j.ipm.2010.08.004.

Wu, D., He, D., Ji, H. and Grishman, R., 2008. A study of using an out-of-box commercial MT system for query translation in CLIR. In: Proceedings of the ACM CIKM Workshop on Improving non-English Web Searching (ACM iNEWS'08). New York: ACM, pp. 71-76.

Appendix A: Description of the tags

This appendix describes the tags that appear in Figure 2 and Figure 3:

Afs	Adjective: feminine, singular
Ams	Adjective: masculine, singular
Nfp	Noun: feminine, plural
Nfs	Noun: feminine, singular
Nms	Noun: masculine, singular
Vi	Verb: infinitive
V1s	Verb: 1st person, singular
V2s	Verb: 2nd person, singular
V3p	Verb: 3rd person, plural
V3s	Verb: 3rd person, singular