Word normalization in Twitter using finite-state transducers

Jordi Porta and **José Luis Sancho**
Centro de Estudios de la Real Academia Española
c/ Serrano 197-198, Madrid 28002
{porta,sancho}@rae.es

Resumen:
Palabras clave:

Abstract: This paper presents a linguistic approach based on weighted-finite state transducers for the lexical normalisation of Spanish Twitter messages. The system developed consists of transducers that are applied to out-of-vocabulary tokens. Transducers implement linguistic models of variation that generate sets of candidates according to a lexicon. A statistical language model is used to obtain the most probable sequence of words. The article includes a description of the components and an evaluation of the system and some of its parameters.

Keywords: Tweet Messages. Lexical Normalisation. Finite-state Transducers. Statistical Language Models.

1 Introduction

Text messaging (or texting) exhibits a considerable degree of departure from the writing norm, including spelling. There are many reasons for this deviation: the informality of the communication style, the characteristics of the input devices, etc. Although many people consider that these communication channels are "deteriorating" or even "destroying" languages, many scholars claim that even in this kind of channels communication obeys maxims and that spelling is also principled. Even more, it seems that, in general, the processes underlying variation are not new to languages. It is under these considerations that the modelling of the spelling variation, and also its normalisation, can be addressed. Normalisation of text messaging is seen as a necessary preprocessing task before applying other natural language processing tools designed for standard language varieties.

Few works dealing with Spanish text messaging can be found in the literature. To the best of our knowledge, the most relevant and recent published works are Mosquera and Moreda (2012), Pinto et al. (2012), Gomez Hidalgo, Caurcel Díaz, and Iñiguez del Rio (2013) and Oliva et al. (2013).

2 Architecture and components of the system

The system has three main components that are applied sequentially: An analyser performing tokenisation and lexical analysis on standard word forms and on other expressions like numbers, dates, etc.; a component generating word candidates for out-of-vocabulary (OOV) tokens; a statistical language model used to obtain the most likely sequence of words; and finally, a truecaser giving proper capitalisation to common words assigned to OOV tokens.

Freeling (Atserias et al., 2006) with a special configuration designed for this task is used to tokenise the message and identify, among other tokens, standard words forms. The generation of candidates, i.e., the confusion set of an OOV token, is performed by components inspired in other modules used to analyse words found in historical texts, where other kind of spelling variation can be found (Porta, Sancho, and Gómez, 2013). The approach to historical variation was based on weighted finite-state transducers over the tropical semiring implementing linguistically motivated models. Some experiments were conducted in order to assess the task of assigning to old word forms their corresponding modern lemmas. For each old word, lemmas were assigned via the possible modern forms predicted by the model. Results were comparable to the results obtained with the Levenshtein distance (Levenshtein, 1966) in terms of recall, but were better in terms of accuracy, precision and F. As for old words, the confusion set of a OOV token is generated by applying the shortest-paths algorithm to the following expression:

$$W \circ E \circ L$$

where W is the automata representing the OOV token, E is an edit transducer generating possible variations on tokens, and L is the set of target words. The composition of these three modules is performed using an on-line implementation of the efficient three-way composition algorithm of Allauzen and Mohri (2008).

3 Resources employed

In this section, the resources employed by the components of the system are described: the edit transducers, the lexical resources and the language model.

3.1 Edit transducers

We follow the classification of Crystal (2008) for texting features present also in Twitter messages. In order to deal with these features several transducers were developed. Transducers are expressed as regular expressions and context-dependent rewrite rules of the form $\alpha \rightarrow \beta \; / \; \gamma \;\_\_\; \delta$ (Chomsky and Halle, 1968) that are compiled into weighted finite-state transducers using the OpenGrm Thrax tools (Tai, Skut, and Sproat, 2011).

3.1.1 Logograms and Pictograms

Some letters are found used as logograms, with a phonetic value. They are dealt with by optional rewrites altering the orthographic form of tokens:

$$\mathit{ReplaceLogograms} = (x \; (\rightarrow) \; \mathit{por}) \circ (\mathit{2} \; (\rightarrow) \; \mathit{dos}) \circ (\mathit{@} \; (\rightarrow) \; a|o) \circ \ldots$$

Also laughs, which are very frequent, are considered logograms, since they represent sounds associated with actions. The multiple ways they are realised, including plurals, are easily described with regular expressions.

Pictograms like emoticons entered by means of ready-to-use icons in input devices are not treated by our system since they are not textual representations. However textual representations of emoticons like *:DDD* or *xDDDDDD* are recognised by regular expressions and mapped to their canonical form by means of simple transducers.

3.1.2 Initialisms, shortenings, and letter omissions

The string operations for initialisms (or acronymisation) and shortenings are difficult to model without incurring in an overgeneration of candidates. For this reason, only common initialisms, e.g., *sq* (*es que*), *tk* (*te quiero*) or *sa* (*se ha*), and common shortenings, e.g., *exam* (*examen*) or *nas* (*buenas*), are listed.

For the omission of letters several transducers are implemented. The simplest and more conservative one is a transducer introducing just one letter in any position of the token string. Consonantal writing is a special case of letter omission. This kind of writing relies on the assumption that consonants carry much more information than vowels do, which in fact is the norm in same languages like Semitic languages. Some rewrite rules are applied to OOV tokens in order to restore vowels:

$$\begin{aligned}&\mathit{InsertVowels} = \mathrm{invert}(\mathit{RemoveVowels})\\&\mathit{RemoveVowels} = \mathit{Vowels} \; (\rightarrow) \; \epsilon\end{aligned}$$

3.1.3 Standard non-standard spellings

We consider non-standard spellings standard when they are widely used. These include spellings for representing regional or informal speech, or choices sometimes conditioned by input devices, as non-accented writing. For the case of accents and tildes, they are restored using a cascade of optional rewrite rules like the following:

$$\mathit{RestoreAccents} = (n|ni|ny|nh \; (\rightarrow) \; \tilde{n}) \circ (a \; (\rightarrow) \; \acute{a}) \circ (e \; (\rightarrow) \; \acute{e}) \circ \ldots$$

Also words containing k instead of c or qu, which appears frequently in protest writings, are standardised with simple transducers. Some other changes are done to some endings to recover the standard ending. There are complete paradigms like the following, which relates non-standard to standard endings:

-a	-ada
-as	-adas
-ao	-ado
-aos	-ados

We also consider phonetic writing as a kind of non-standard writing in which a phonetic form of a word is alphabetically and syllabically approximated. The transducers used for generating standard words from their phonetic and graphical variants are:

$$\textit{DephonetiseWriting} = \text{invert}(\textit{PhonographemicVariation})$$

$$\textit{PhonographemicVariation} = \textit{GraphemeToPhoneme} \circ \textit{PhoneConflation} \circ \textit{PhonemeToGrapheme} \circ \textit{GraphemeVariation}$$

In the previous definitions, the *PhoneConflation* makes phonemes equivalent, as for example the IPA phonemes /ʎ/ and /ɟ/. Linguistic phenomena as *seseo* and *ceceo*, in which several phonemes were conflated by 16th century, still remain in spoken variants and are also reflected in texting. The *GraphemeVariation* transducer models, among others, the writing of *ch* as *x*, which could be due to the influence of other languages.

3.1.4 Juxtapositions

Spacing in texting is also non-standard. In the normalisation task, some OOV tokens are in fact juxtaposed words. The possible decompositions of a word into a sequence of possible words is: shortest-paths($W \circ$ *SplitConjoinedWords* $\circ\, L(\_L)^{+}$), where W is the word to be analysed, $L(\_L)^{+}$ represents the valid sequences of words and *SplitConjoinedWords* is a transducer introducing blanks (_) between letters and undoing optionally possible fused vowels:

$$\textit{SplitConjoinedWords} = \text{invert}(\textit{JoinWords})$$

$$\textit{JoinWords} = (a\_a\ (\rightarrow)\ a{<}1{>}) \circ \ldots \circ (u\_u\ (\rightarrow)\ u{<}1{>}) \circ (\_\ (\rightarrow)\ \epsilon)$$

Note that in the previous definition, some rules are weighted with a unit cost <1>. These costs are used by the shortest-paths algorithm as a preference mechanism to select non-fused over fused sequences when both cases are possible.

3.1.5 Other transducers

Expressive lengthening, which consist in repeating a letter in order to convey emphasis, are dealt with by means of rules removing a varying number of consecutive occurrences of the same letter. An example of a rule dealing with letter a repetitions is $a\ (\rightarrow)\ \epsilon\ /\ a$ ___ . A transducer is generated for the alphabet.

Because messages are keyboarded, some errors found in words are due to letter transpositions and confusions between adjacent letters in the same row of the keyboard. These changes are also implemented with a transducer.

Finally, a Levenshtein transducer with a maximum distance of one has been also implemented.

3.2 The lexicon

The lexicon for OOV token normalisation contains mainly Spanish standard words, proper names and some frequent English words. These constitute the set of target words. We used the DRAE (RAE, 2001) as the source for Spanish standard words in the lexicon. Besides inflected forms, we have added verbal forms with clitics attached and derivative forms not found as entries in the DRAE: *-mente* adverbs, appreciatives, etc. The list of proper names was compiled from many sources and contains first names, surnames, aliases, cities, country names, brands, organisations, etc. Special attention was payed to hypocorisms, i.e., shorter or diminutive forms of a given name, as well as nicknames or calling names, since communication in channels as Twitter tends to be interpersonal (or between members of a group) and affective. A list of common hypocorisms is provided to the system. For English words, we have selected the 100,000 more frequent words of the BNC (BNC, 2001).

3.3 Language model

We use a language model to decode the word graph and thus obtain the most probable word sequence. The model is estimated from a corpus of webpages compiled with Wacky (Baroni et al., 2009). The corpus contains about 11,200,000 tokens coming from about 21,000 URLs. We used as seeds the types found in the development set (about 2,500). Backed-off n-gram models, used as language models, are implemented with the OpenGrm NGram toolkit (Roark et al., 2012).

3.4 Truecasing

The restoring of case information in badly-cased text has been addressed in (Lita et al., 2003) and has been included as part of the normalisation task. Part of this process, for proper names, is performed by the application of the language model to the word graph. Words at message initial position are not always uppercased, since doing so yielded contradictory results after some experimentation. A simple heuristic is implemented to uppercase a normalisation candidate when the OOV token is also uppercased.

4 Settings and evaluation

In order to generate the confusion sets we used two edit transducers applied in a cascade. If neither of the two is able to relate a token with a word, the token is assigned to itself.

The first transducer generates candidates according to the expansion of abbreviations, the identification of acronyms, pictograms and words which result from the following composition of edit transducers combining some of the features of texting:

$$
\begin{array}{l}
\mathit{RemoveSymbols} \circ \\
\mathit{LowerCase} \circ \\
\mathit{Deaccent} \circ \\
\mathit{RemoveReduplicates} \circ \\
\mathit{ReplaceLogograms} \circ \\
\mathit{StandardiseEndings} \circ \\
\mathit{DephonetiseWriting} \circ \\
\mathit{Reaccent} \circ \\
\mathit{MixCase}
\end{array}
$$

The second edit transducer analyses tokens that did not receive analyses with the first editor. This second editor implements consonantal writing, typing error recovery, an approximate matching using a Levenshtein distance of one and the splitting of juxtaposed words. In all cases, case, accents and reduplications are also considered. This second transducer makes use of an extended lexicon containing sequences of simple words.

Several experiments were conducted in order to evaluate some parameters of the system. In particular, the effect of the order of the n-grams in the language model and the effect of generating confusion sets for OOV tokens only versus the generation of confusion sets for all tokens. For all the experiments we used the test set provided with the tokenization delivered by Freeling.

For the first series of experiments, tokens identified as standard words by Freeling receive the same token as analysis and OOV tokens are analysed with the system. Recall on OOV tokens is of 89.40 %. Confusion sets size follows a power-law distribution with an average size of 5.48 for OOV tokens that goes down to 1.38 if we average over the rest of the tokens. Precision for 2- and 4-gram language models is 78.10 %, but the best result is obtained with 3-grams, with an precision of 78.25 %.

There is a number of non-standard forms that were wrongly recognised as in-vocabulary words because they clash with other standard words. In the second series of experiments a confusion set is generated for each word in order to correct potentially wrong assignments. Average size of confusion sets increases to 5.56.[1] Precision results for the 2-gram language model is of 78.25 % but 3- and 4-gram reach both an precision of 78.55 %.

From a quantitative point of view, it seems that slighty better results are obtained using a 3-gram language model and generating confusion sets not only for OOV tokens but for all the tokens in the message. In a qualitative evaluation of errors several categories show up. The most populated categories are those having to do with case restoration and wrong decoding by the language model. Some errors are related to particularities of DRAE, from which the lexicon was derived (*dispertar* or *malaleche*). Non standard morphology is observed in tweets, as in derivatives (*tranquileo* or *loquendera*). Lack of abbreviation expansion is also observed (*Hum*). Faulty application of segmentation accounts for a few errors (*mencantaba*). Finally, some errors are not on our output but on the reference (*Hojo*).

5 Conclusions and future work

No attention has been payed to multilingualism since the task explicitly excluded tweets from bilingual areas of Spain. However, given that not few Spanish speakers (both in Europe and America) are bilingual or live in bilingual areas, mechanisms should be provided to deal with other languages than English to make the system more robust.

We plan to build a corpus of lexically standard tweets via the Twitter streaming API to determine whether n-grams observed in a

[1] We removed from the calculation the token *mes_de_abril*, which receives 308,017 different analyses due to the combination of multiple editions and segmentations.

Twitter-only corpus improve decoding or not as a side effect of syntax being also non standard.

Qualitative analysis of results showed that there is room for improvement experimenting with selective deactivation of items in the lexicon and further development of the segmenting module.

However, initialisms and shortenings are features of texting difficult to model without causing overgeneration. Acronyms like *FYQ*, which correspond to the school subject of *Física y Química*, are domain specific and difficult to foresee and therefore to have them listed in the resources.

References

Allauzen, Cyril and Mehryar Mohri. 2008. 3-way composition of weighted finite-state transducers. In *Proc. of the 13th Int. Conf. on Implementation and Application of Automata (CIAA–2008)*, pages 262–273, San Francisco, California, USA.

Atserias, Jordi, Bernardino Casas, Elisabet Comelles, Meritxell González, Lluís Padró, and Muntsa Padró. 2006. FreeLing 1.3: Syntactic and semantic services in an open-source NLP library. In *Proc. of the 5th Int. Conf. on Language Resources and Evaluation (LREC-2006)*, pages 48–55, Genoa, Italy, May.

Baroni, Marco, Silvia Bernardini, Adriano Ferraresi, and Eros Zanchetta. 2009. The wacky wide web: A collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, 43(3):209–226.

BNC. 2001. The British National Corpus, version 2 (BNC World). Distributed by Oxford University Computing Services on behalf of the BNC Consortium. http://www.natcorp.ox.ac.uk.

Chomsky, Noam and Morris Halle. 1968. *The sound pattern of English.* Harper & Row, New York.

Crystal, David. 2008. *Txtng: The Gr8 Db8.* Oxford University Press.

Gomez Hidalgo, José María, Andrés Alfonso Caurcel Díaz, and Yovan Iñiguez del Rio. 2013. Un método de análisis de lenguaje tipo SMS para el castellano. *Linguamática*, 5(1):31—39, July.

Levenshtein, Vladimir I. 1966. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, 10(8):707–710.

Lita, Lucian Vlad, Abe Ittycheriah, Salim Roukos, and Nanda Kambhatla. 2003. tRuEcasIng. In *Proc. of the 41st Annual Meeting on ACL - Volume 1*, ACL '03, pages 152–159, Stroudsburg, PA, USA.

Mosquera, Alejandro and Paloma Moreda. 2012. TENOR: A lexical normalisation tool for Spanish Web 2.0 texts. In Petr Sojka, Aleš Horák, Ivan Kopeček, and Karel Pala, editors, *Text, Speech and Dialogue*, volume 7499 of *LNCS*. Springer, pages 535–542.

Oliva, J., J. I. Serrano, M. D. Del Castillo, and Á. Iglesias. 2013. A SMS normalization system integrating multiple grammatical resources. *Natural Language Engineering*, 19:121–141, 1.

Pinto, David, Darnes Vilariño Ayala, Yuridiana Alemán, Helena Gómez, Nahun Loya, and Héctor Jiménez-Salazar. 2012. The Soundex phonetic algorithm revisited for SMS text representation. In Petr Sojka, Aleš Horák, Ivan Kopeček, and Karel Pala, editors, *Text, Speech and Dialogue*, volume 7499 of *LNCS*, pages 47–55. Springer.

Porta, Jordi, José-Luis Sancho, and Javier Gómez. 2013. Edit transducers for spelling variation in Old Spanish. In *Proc. of the workshop on computational historical linguistics at NODALIDA 2013. NEALT Proc. Series 18*, pages 70–79, Oslo, Norway, May 22-24.

RAE. 2001. *Diccionario de la lengua española.* Espasa, Madrid, 22th edition.

Roark, Brian, Richard Sproat, Cyril Allauzen, Michael Riley, Jeffrey Sorensen, and Terry Tai. 2012. The OpenGrm open-source finite-state grammar software libraries. In *Proc. of the ACL 2012 System Demonstrations*, pages 61–66, Jeju Island, Korea, July.

Tai, Terry, Wojciech Skut, and Richard Sproat. 2011. Thrax: An Open Source Grammar Compiler Built on OpenFst. In *ASRU 2011*, Waikoloa Resort, Hawaii, December.