Spelling Correction: from two-level morphology to open source

Iñaki Alegria, Klara Ceberio, Nerea Ezeiza, Aitor Soroa, Gregorio Hernandez
Ixa group. University of the Basque Country / Eleka S.L.
649 P.K. 20080 Donostia. Basque Country.
i.alegria@ehu.es

Abstract

Basque is a highly inflected and agglutinative language (Alegria *et al*., 1996). Two-level morphology has been applied successfully to this kind of languages and there are two-level based descriptions for very different languages. After doing the morphological description for a language, it is easy to develop a spelling checker/corrector for this language. However, what happens if we want to use the speller in the "free world" (*OpenOffice*, *Mozilla*, *emacs*, *LaTeX*, ...)? *Ispell* and similar tools (*aspell*, *hunspell*, *myspell*) are the usual mechanisms for these purposes, but they do not fit the two-level model. In the absence of two-level morphology based mechanisms, an automatic conversion from two-level description to *hunspell* is described in this paper.

1. Introduction

Two-level morphology (Koskenniemi, 1983; Beesley & Karttunen, 2003) has been applied successfully to the morphological description of highly inflected languages. There are two-level based descriptions for very different languages (English, German, Swedish, French, Spanish, Danish, Norwegian, Finnish, Basque, Russian, Turkish, Arab, Aymara, Swahili, etc.).

After doing the morphological description, it is easy to develop a spelling checker/corrector for the language (Kukich, 1992). The spelling checker will accept as correct any word which allows a correct standard morphological breakdown. When a word is not recognised by the checker, it is assumed to be a misspelling. For the correction Damerau's classification (edit distance of one) is used in order to generate hypothetical corrections, and those which are accepted by the spelling checker will be displayed (Aldezabal *et al*., 1999)

From 1992, there is a spelling checker/corrector for Basque (Alegria *et al*., 1996) using this approach (Aduriz *et al*., 1997) with different versions (for different text editors, web and OCR[1]). Nevertheless, when we wanted to use this approach in the GNU/*Linux* world we had three main choices:

1. To use *ispell* and similar tools (*aspell*, *hunspell*, *myspell*) in order to catch the maximum number of applications.
2. To adapt our implementation to each possible application
3. To propose and implement a new tool based on the two-level morphology, in a coordinated initiative with other partners.

Even though the third option is very interesting, we decided to face the first one but in an automatic way. We wanted to adapt the two-level morphology for the new purposes. In our opinion, this is an interesting approach for highly inflected languages with two-level description for morphology, which can to take advantage of the previous work and reuse the morphological information.

[1] www.xuxen.com

2. Free software for spelling correction

Unfortunately there are not open source tools for spelling correction with these features:

- It is standardized in the most of the applications (*OpenOffice*, *Mozilla*, *emacs*, *LaTeX*, ...).
- It is based on the two-level morphology.

The *spell* family of spell checkers (*ispell*, *aspell*, *myspell*) fits the first condition but not the second. The description of the lemmas (or stems) and affixes have to be carried out without distinguishing morphotactical and phonological phenomena. It would be suitable for our aims if it would be adequate for highly inflected languages.

Moreover, *ispell*, the oldest and the most widespread tool, is quite limited to be applied to agglutinative and highly inflected languages. Only sixty four paradigms can be defined and it is not possible to link new morphemes after the suffixes. *Myspell* is a new C++ implementation of *ispell*, and *aspell* does not improve the description power, it is oriented to obtain good proposals for correction.

Hunspell (Nemeth *et al*., 2004) is an improvement of *myspell* to face our problem. It is more expressive than *ispell* because it is possible to define more paradigms and to chain two suffixes. Nevertheless, it is not two-level based, and phonology and morphotactics are described together like in *ispell* or *myspell*. *Hunspell* has been successful and it has been adopted as standard for the new versions of *OpenOffice* and *Mozilla* (*Firefox* and *Thunderbird*).

3. From two-level description to *hunspell*

The simplest way of making a spelling checker/corrector is to build a very large list of correct words and integrating them in *aspell*. Additionally, it is possible to combine few paradigms with a list of the forms for the uncovered words in order to make an *ispell* compliant description, but two big problems arise in this approach:

- The list can be too large, or, if the list is limited, the coverage will be low.

- The tool lacks coherency. Given a lemma, the wordforms corresponding to some declensions are accepted and other wordforms are rejected. We think this is an important problem, because the tool loses the trainer profile.

Our solution was to adapt the two-level description to *hunspell* in a (semi)automatic way. We propose an eight steps procedure to address the problem and we have applied it for the Basque.

We will present the procedure step by step.

Firstly, we want to underline that the morphotactical description for Basque was made in a recursive way, allowing to link suffixes one after the other, following some constraints. But this description was unsuitable for our goal.

Thus, in the first step, we have constrained the morphotactical description for Basque. For this purpose, we have described a couple of new two-level rules in order to limit the number of linked suffixes. After different experiments using a very large list of correct words, we decided to limit the number of suffixes in a word to three, with some exceptions (plural and nominalization suffixes are not counted).

The (simplified) main rule for this purpose is the following:

```
# MM morpheme border
# Ch other characters
# GEN person genitive
# GEL place genitive
MM:MM /<= MM GEN (MM Ch+)+ (MM) _ GEN ;
     # two person genitives not allowed
     MM GEL (MM Ch+)+ (MM) _ GEL ;
     # two place genitives not allowed
     MM Ch+  MM Ch+ MM Ch+ _ ;
     # no more than 3 suffixes
```

The second step was devoted to building a new two-level system that used these constraint rules. This system is used in other steps and it is useful to test the final result. It is important to point out that this new system and the *hunspell* compliant system must be equivalent; when one of the systems accepts/rejects a word, the other one must do the same.

In the third step, we wanted to face the phonological transformations that occur in the boundaries of connected morphemes. Thus, we revised the phonological rules to obtain the endings of lemmas which can change when linked to suffixes. We used this list of endings and the list of lemmas to build a new list of phonologically classified lemmas (and stems) for each paradigm, and we selected one representative for each class. The selected endings of lemmas were: a, e, r, l, m, n, t, k, z, s, x, tz, tx and ts. All of them have interesting phonological features which are described by two-level rules in the original description..

This is the weakest side of our method for two reasons: it is a manual process and it cannot be applied to more complex phonological phenomena as vowel harmony in Finnish.

The fourth step, we generated all the possible wordforms for each lemma and stem selected in the previous step. For this purpose, we applied the morphological analyzer/generator obtained in the second step.

Afterwards, we had all the possible wordforms and we wanted to obtain a list of endings to use it for ispell/myspell. We reduced each word to the endings (suffix or set of linked suffixes) based on the lemmas and stems used in the previous step. However, there are two problems to use this information directly for the named applications: there are too many paradigms and the lists are too large. Therefore, we decided to transform these endings to obtain a more compact description.

In the 6th step, we perform this transformation dividing the endings in two pieces: left-side and right-side. Our strategy has been to find a genitive morpheme and cut after it. The left-pieces remained in the paradigm and we built new paradigms with the right-pieces, the second level paradigms.

It could seem that there will be a large amount of new paradigms in this second level. However, in the 7th step, we applied a minimization process and reduce them to a small amount. This minimization process is very simple, we collapse identical paradigms.

After the 6th and 7th steps, the number of paradigms is incremented from 360 to 710 (350 paradigms in the second level), but the length of the whole description decreases from 375,000 lines to 165,000.

Finally, we added all the original stems along with their paradigms identification to obtain the complete description.

4. Results and evaluation

The final result of the conversion process is all the information we need for the *hunspell* description: the stems and two sets of suffixes corresponding to the paradigms at first and second level respectively. Only a format conversion is necessary to deliver the spelling checker/corrector for *OpenOffice* and other tools integrating *hunspell*.

The generated morphological description has been reused, including the morphological information corresponding to the morphemes, for the morphological generation in a machine translation engine, named *Matxin* (Alegria *et al*., 2007). This tool is integrated in the *OpenTrad*[2] initiative whose main features are interoperability, standardization and free software.

In addition, we did the adaptation of the description to *myspell*, for tools not integrating *hunspell*, combining the main paradigms (with less generation power for each one) and the wordforms appearing in a big corpus, after eliminating forms rejected by the original spelling checker. In this case the mentioned inconveniences regarding to the coverage and the lack of coherence appeared.

These resources, under GPL licences, are publicly

[2] www.opentrad.org

available in the following URLs:

- *hunspell*: www.euskara.euskadi.net
- *myspell*; www.librezale.org/mozilla/firefox
- *opentrad-matxin*: matxin.sourceforge.net

The evaluation was carried out comparing the results using the speller based on the two-level description and the *hunspell* speller. Using a big corpus with more than 20,000 different words, we only detected 112 disagreements (all of them false misspellings of *hunspell*) were detected. 68 of them were correct words which were not recognized by *hunspell* because the morphotactics are constrained (step 1). The other 44 differences are real errors which were not detected in the two-level description because of the overgeneration of the description.

We present some examples of the speller. In Fig. 1, there is a caption of *OpenOffice* where the word "erabiltaileen" is underlined and we can see the correction proposal "erabiltzaileen" ("*of the users*"), which is the correct spelling of the word. In the second example, we can see the application of *ispell* in *Mozilla Thunderbird*, correcting an email will it is composed.

5. Conclusions

We have presented a procedure to migrate from a two-level morphological description to open-source spelling correction. This solution has been carried out for Basque, but we think it is general enough and it can be used for the conversion of morphological descriptions of other languages.

However, the adopted solution using *hunspell* has two limitations:

1. The method has a manual step (the third step in section 3), which is simple but difficult to automatize. So, for the moment, a linguist has to interpret the two-level phonological rules and extract the endings that can change when linked to suffixes. Furthermore, for some languages as Finnish, this can be unfeasible due to the wide range of the phonological phenomena to take into account.
2. The result does not take advantage of the changes when linking affixes that can be described using *hunspell*. So, it would be possible to reduce the description collapsing equivalent paradigms after changing, eliminating or adding a character on the left of the suffixes.

A deeper improvement would deal with improving *hunspell* in order to manage more than two linked suffixes. Even so, we still think that the most adequate solution would be to have a general mechanism available, equivalent to *hunspell*, interpreting directly morphological descriptions based on two-level morphology.

6. Acknowledgements

This work has been partially by the Basque Government (Department of Language Policy and Department of Industry –Anhitz project, Etortek IE06-185–).

7. References

Aduriz I., Alegria I., Artola X., Ezeiza N., Sarasola K. (1997) A spelling corrector for Basque based on morphology. *Literary & Linguistic Computing, Vol. 12, No. 1*. Oxford University Press. Oxford..

Aduriz I., Aldezabal I., Alegria I., Arriola J., Díaz de Ilarraza A., Ezeiza N., Gojenola K. (2003) Finite State Applications for Basque *Proc. of EACL'2003* Workshop on Finite-State Methods in Natural Language Processing

Aldezabal I., Alegria I., Ansa O., Arriola J., Ezeiza N. (1999). Designing spelling correctors for inflected languages using lexical transducers. *Proceedings of EACL'99*. pg. 265-266. Bergen, Norway.

Alegria I., Artola X., Sarasola K., Urkia M. (1996). Automatic morphological analysis of Basque. *Literary & Linguistic Computing* Vol. 11, No. 4, 193-203. Oxford University Press. Oxford.

Alegria I., Díaz de Ilarraza A., Labaka G., Lersundi M., Mayor A., Sarasola K. (2007). Transfer-based MT from Spanish into Basque: reusability, standardization and open source. LNCS 4394. Cicling 2007.

Beesley K.R. and Karttunen L. (2003) *Finite State Morphology*. CSLI Publications, Palo Alto, CA.

Koskenniemi, K. (1983) *Two-level morphology: A general computational model of word-form recognition and production*. Tech. Rep. Publication No. 11, Dept. of General Linguistics, University of Helsinki.

Kukich K. (1992). Techniques for automatically correcting word in text. *ACM Computing Surveys*, vol.24, No. 4, 377-439

Nemeth V., Tron P., Halacsy A., Kornai A., Rung I. (2004) Leveraging the open source ispell codebase for minority language analysis. *Proceedings of SALTMIL*.

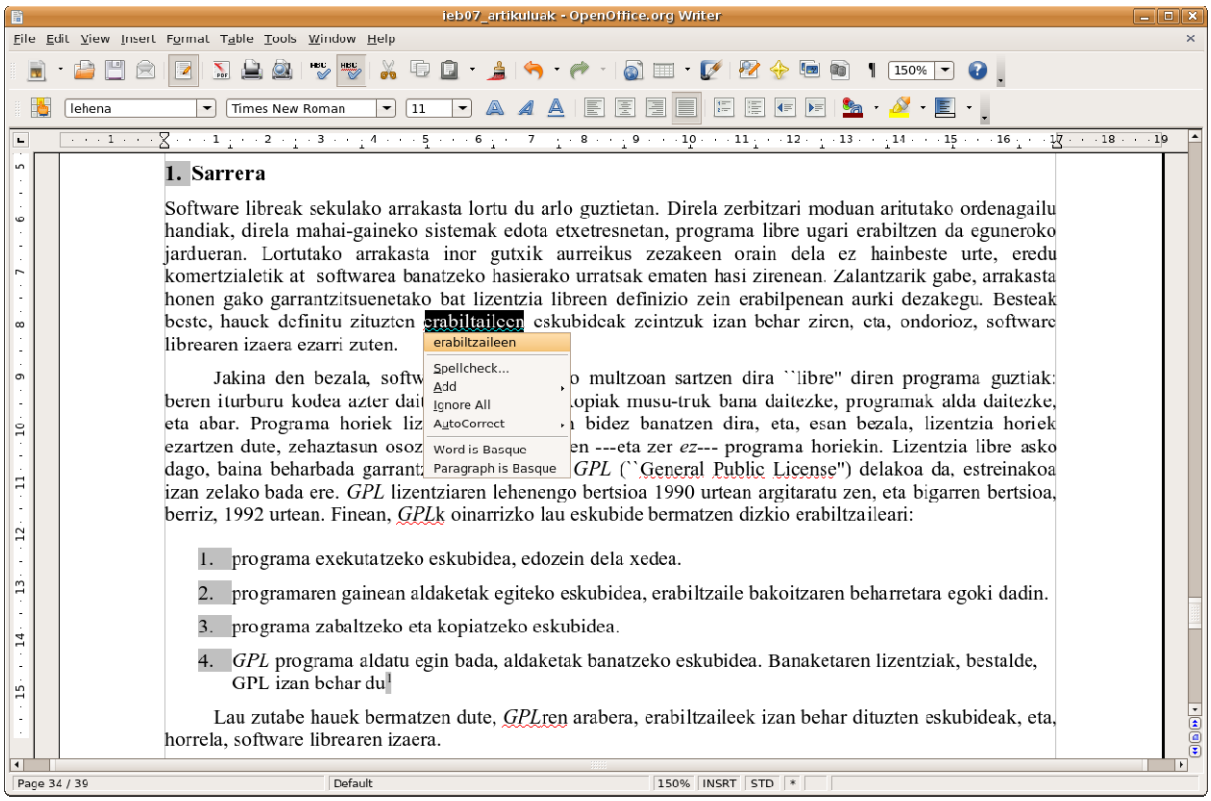

Fig. 1- Using *hunspell* for Basque in *OpenOffice2*

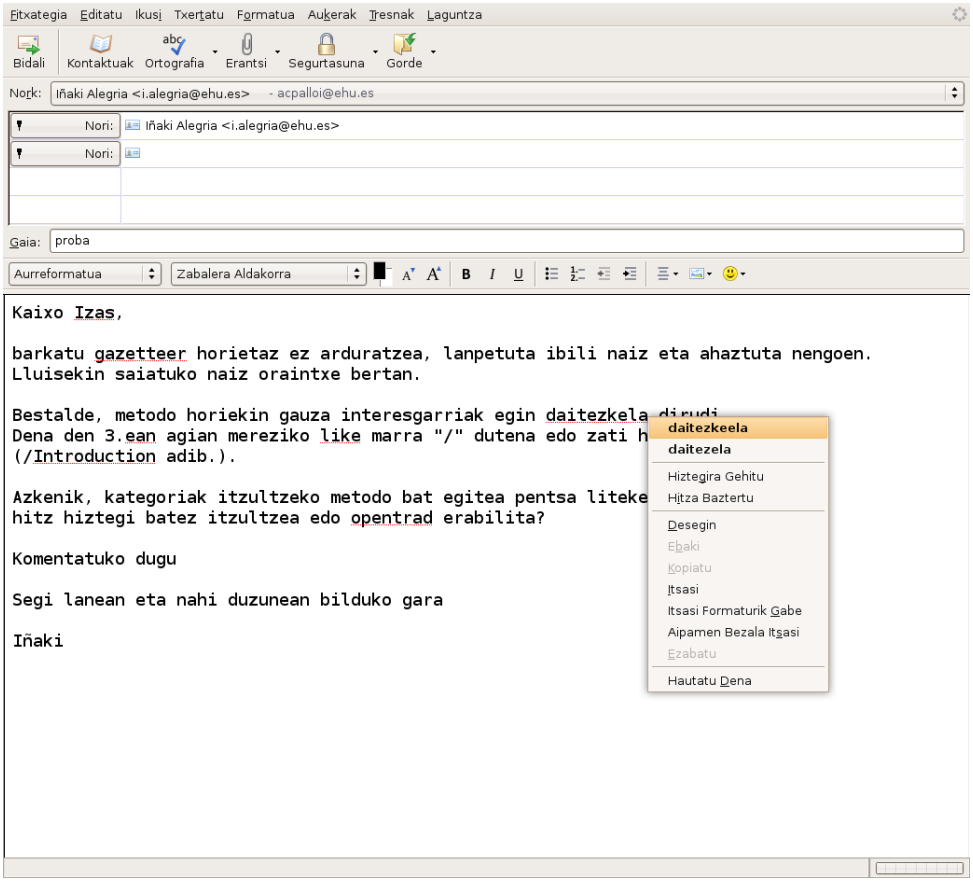

Fig. 2- Using *ispell* for Basque in *Mozilla Thunderbird*