An integrated tool for annotating historical corpora

Pablo Picasso Feliciano de Faria* University of Campinas, Campinas, Brazil, pablofaria@gmail.com
Fabio Natanael Kepler† University of Sao Paulo, Sao Paulo, Brazil, kepler@ime.usp.br
Maria Clara Paixão de Sousa University of Sao Paulo, Sao Paulo, Brazil, mclara.ps@gmail.com

Abstract

E-Dictor is a tool for encoding, applying levels of editions, and assigning part-of-speech tags to ancient texts. In short, it works as a WYSIWYG interface to encode text in XML format. It comes from the experience during the building of the Tycho Brahe Parsed Corpus of Historical Portuguese and from consortium activities with other research groups. Preliminary results show a decrease of at least 50% on the overall time taken on the editing process.

1 Introduction

The Tycho Brahe Parsed Corpus of Historical Portuguese (CTB) (Cor, 2010) consists of Portuguese texts written by authors born between 1380 and 1845. Been one of the forefront works among projects dedicated to investigate the history of Portuguese language, it contributed to the renovation of the theoretical relevance of studies about the linguistic change in different frameworks (Mattos e Silva, 1988; Kato and Roberts, 1993; de Castilho, 1998).

This resulted in crescent work with ancient texts in the country (Megale and Cambraia, 1999), and, by the end of the 1990s, the work on Corpus Linguistics has given rise to a confluence between philology and computer science, a relationship not so ease to equate.

1.1 Philological and computational needs

In studies based on ancient texts, above all, one has to guarantees fidelity to the original forms of the texts. Starting with a *fac-simile*, a first option would be the automatic systems of character recognition (OCR). For the older texts, however, the current recognition technologies have proven inefficient and quite inadequate for handwritten documents (Paixão de Sousa, 2009). Anyway one cannot totally avoid manual transcription.

There are different degrees of fidelity between the transcription and the original text. In practice, one often prepares a "semi-diplomatic" edition, in which a slightly greater degree of interference is considered acceptable – eg., typographical or graphematic modernization. A central goal of the philological edition is to make the text accessible to the specialist reader, with maximum preservation of its original features.

However, it needs to be integrated with computational and linguistic requirements: the need for quantity, agility and automation in the statistical work of selecting data. The original spelling and graphematic characteristics of older texts, for example, may hinder the subsequent automatic processing, such as morphological annotation. Thus, the original text needs to be prepared, or edited, with a degree of interference higher than that acceptable for a semi-diplomatic edition and that is where the conflict emerges.

1.2 Background

The modernization of spellings and standardization of graphematic aspects, during the first years of CTB, made texts suitable for automated processing, but caused the loss of important features from the original text for the historical study of language. This tension has led to the project "Memories of the Text" (Paixão de Sousa, 2004), which sought to restructure the Corpus, based on the development of XML annotations (W3C, 2009), and to take advantage of the core features of this type of encoding, for example, XSLT (W3C, 1999) processing.

A annotation system was conceived and applied to 48 Portuguese texts (2,279,455 words), which

* Thanks to FAPESP, n. 2008/04312-9, for funding part of the development of E-Dictor.
† Thanks to CAPES for the scholarship granted during the initial part of this work.

Proceedings of the Fourth Linguistic Annotation Workshop, ACL 2010, pages 217–221,
Uppsala, Sweden, 15-16 July 2010.

allowed keeping philological informations while making the texts capable of being computationally treated in large-scale. Since 2006, the system has been being tested by other research groups, notably the *Program for the History of Portuguese Language* (PROHPOR-UFBA). The system, then, met its initial objectives, but it had serious issues with respect to reliability and, especially, ease of use.

We noted that manual text markup in XML was challenging to some and laborious for everyone. The basic edition process was: transcription in a text editor, application of the XML markup (tags plus philological edition), generation of a standardized plain text version to submit to automatic part-of-speech tagging, revision of both files (XML and tagged). All in this process, except for text tagging, been manually done, was too subject to failures and demanded constant and extensive revision of the encoding. The need for an alternative, to make the task more friendly, reliable, and productive, became clear. In short, two things were needed: a friendly interface (WYSIWYG), to prevent the user from dealing with XML code, and a way to tighten the whole process (transcription, encode/edition, POS tagging and revision).

1.3 Available tools

A search for available options in the market (free and non-free) led to some very interesting tools, which may be worth trying:

- *Multext*[1]: a series of projects for corpora encoding as well as developing tools and linguistic resources. Not all tools seem to have been finished, and the projects seems to be outdated and no longer being maintained.
- *CLaRK*[2]: a system for corpora development based on XML and implemented in Java. It does not provide a WYSIWYG interface.
- *Xopus*[3]: an XML editor, which offers a WYSIWYG interface. Some of its funcionalities can be extended (customized) throught a Javascript API.
- *<oXygen/> XML Editor*[4]: a complete XML development platform with support for all major XML related standards. An XML file can be edited in the following perspectives: XML text editor, WYSIWYG-like editor, XML grid editor, tree editor.

Unfortunately, all the cited tools lack the capability of dealing proper with *levels of edition* for tokens (words and punctuations) and an integrated environment for the whole process of edition. Thus, in spite of their amazing features, none of them was sufficiently suitable, specially concerning spelling modernization and normalization of graphematic aspects. In fact, this is expected for the tools are intended to broader purposes.

1.4 Solution

Conception and development of a tool, E-Dictor, where the need for a WYSIWYG interface joined a second goal, ie., integrating the tasks of the whole process, which would then be performed inside the same environment, with any necessary external tools being called by the system, transparently.

2 Integrated annotation tool

2.1 General features

E-Dictor has been developed in Python[5] and, today, has versions for both Linux and Windows (XP/Vista/7) platforms. A version for MacOS is planned for the future. It is currently at 1.0 *beta* version (not stable).

2.2 General interface features

As shown in Figure 1, the main interface has an application menu, a toolbar, a content area (divided into tabs: *Transcription*, *Edition*, and *Morphology*), and buttons to navigate throught pages. The tabs are in accordance with the flow of the encoding process. Many aspects of the functioning described in what follows are determined by the application preferences.

In the 'Transcription' tab, the original text is transcribed "as is" (the user can view the fac-simile image, while transcribing the text). Throught a menu option, E-Dictor will automatically apply an XML structure to the text, "guessing" its internal structure as best as it can. Then, in the 'Edition' tab, the user can edit any token or

[1] http://aune.lpl.univ-aix.fr/projects/multext/.
[2] http://www.bultreebank.org/clark.
[3] http://xopus.com/.
[4] http://www.oxygenxml.com/.

[5] Available on internet at http://www.python.org/, last access on Jan, 21th, 2010. Python has been used in a number of computational linguistics applications, e.g., the *Natural Language Toolkit* (Bird et al., 2009).

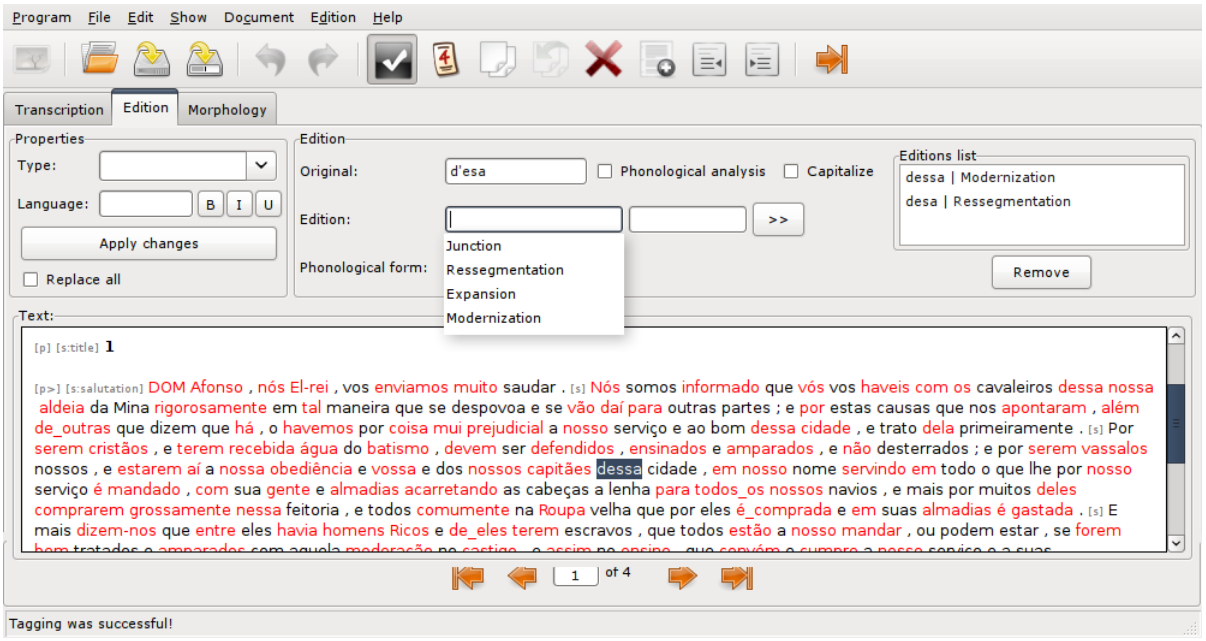

Figure 1: E-Dictor GUI.

structural element (eg., paragraph). Finally, in the 'Morphology' tab, tokens and part-of-speech tags are displayed in token/TAG format, so they can be revised[6].

2.3 The XML structure

The XML structure specified meets two main goals: (i) be as neutral as possible (in relation to the textual content encoded) and (ii) suit philological and linguistic needs, i.e., edition must be simple and efficient without losing information relevant to philological studies. In the context of CTB, it was initially established a structure to encode the following information:

- Metadata: information about the source text, e.g., author information, state of processing, etc.
- Delimitation of sections, pages, paragraphs, sentences, headers and footers, and tokens.
- Class of tokens (part-of-speech tags) and phonological form for some tokens.
- Types (levels) of edition for each token.
- Comments of the editor.
- Subtypes for some text elements, like sections, paragraphs, sentences and tokens (eg., a section of type "prologue").

[6]The current version of E-Dictor comes with a POS tagger, developed by Fabio Kepler, accessed by a menu option.

2.4 Encoding flexibility

A key goal of E-Dictor is to be flexible enough so as to be useful in other contexts of corpora building. To achieve this, the user can customize the "preferences" of the application. The most prominent options are the *levels of edition* for tokens; the subtypes for the elements 'section', 'paragraph', 'sentence', and 'token'; and the list of POS tags to be used in the morphological analysis. Finally, in the 'Metadata' tab, the user can create the suitable metadata fields needed by his/her project.

2.5 Features

Throught its menu, E-Dictor provides some common options (eg., Save As, Search & Replace, Copy & Paste, and many others) as well as those particular options intended for the encoding process (XML structure generation, POS automatic tagging, etc.). E-Dictor provides also an option for exporting the encoded text and the *lexicon of editions*[7] in two different formats (HTML and TXT/CSV).

2.6 Edition

To conclude this section, a brief comment about token (words and punctuation) edition, which is the main feature of E-Dictor. The respective interface is shown in Figure 2. When a token is se-

[7]The actual editions applied to words and punctuations of the original text.

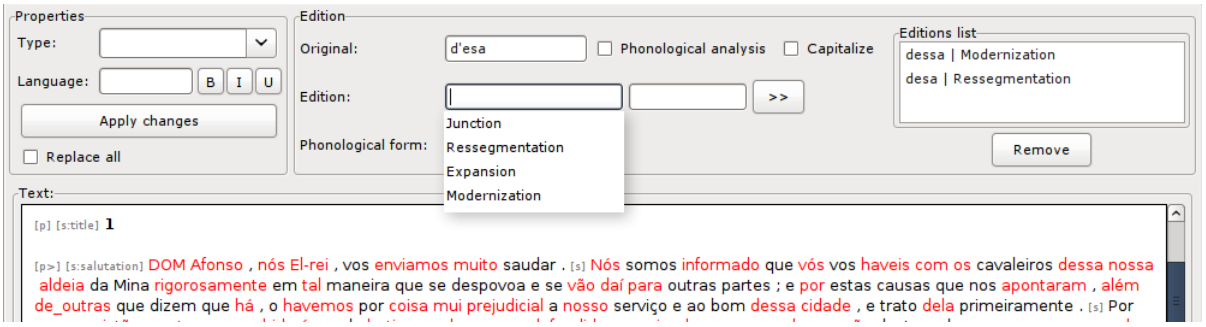

Figure 2: Details of the token edition interface.

lected, the user can: (i) in the "Properties" panel, specify the *type* of the token (according to the subtypes defined by the preferences), its foreign *language*, and *format* (bold, italic, and underlined); (ii) in the "Edition" panel, specify some other properties (eg., phonological form) of the token and include *edition levels* (according to the levels defined by the preferences).

To each token, the user must click on "Apply changes" to effectivate (all) the editions made to it. The option "Replace all" tells E-Dictor to repeat the operation over all identical tokens in the remaining of the text (a similar functionality is available for POS tags revision).

```
<sc id="sc_1">
  <p id="p_1">
    <s id="s_1">
      <w id="s_1#0">
        <o>Ex.mo</o>
        <e t="mod">Excelentíssimo</e>
        <e t="exp">Excelentissimo</e>
      </w>
      <w id="s_1#1">
        <o>Sr.</o>
        <e t="exp">Senhor</e>
      </w>
      <w id="s_1#2">
        <o>Duque</o>
      </w>
```

Figure 3: Example of XML textual encoding.

3 Discussion

The dificulties of encoding ancient texts in XML, using common text editors, had shown that a tool was necessary to make the process efficient and friendly. This led to the development of E-Dictor, which, since its earlier usage, has shown promising results. Now, the user does not even have to know that the underlying encoding is XML. It is only necessary for him/her to know the (philological and linguistics) aspects of text edition.

E-Dictor led to a decrease of about 50% in the time required for encoding and editing texts. The improvement may be even higher if we consider the revision time. One of the factors for this improvement is the better legibility the tool provides. The XML code is hidden, allowing one to practically read the text without any encoding. To illustrate the opposite, Figure 3 shows the common edition "interface", before E-Dictor. Note that the content being edited is just "Ex.mo Sr. Duque".

Finally, the integration of the whole process into one and only environment is a second factor for the overall improvement, for it allows the user to move freely and quickly between "representations" and to access external tools transparently.

3.1 Improvements

E-Dictor is always under development, as we discuss its characteristics and receive feedback from users. There is already a list of future improvements that are being developed, such as extending the exporting routines, for example. A bigger goal is to incorporate an edition lexicon, which would be used by the tool for making suggestions during the edition process, or even to develop an "automatic token edition" system for later revision by the user.

3.2 Perspectives

Besides CTB, E-Dictor is being used by the BBD project (BBD, 2010), and, recently, by various subgroups of the PHPB project (*For a History of Portuguese in Brazil*). These groups have large experience in philological edition of handwritten documents, and we hope their use of E-Dictor will help us improve it. The ideal goal of E-Dictor is to be capable of handling the whole flow of linguistic and philological tasks: transcription, edition, tagging, *and parsing*.

References

[BBD2010] BBD. 2010. Biblioteca Brasiliana Digital.

[Bird et al.2009] Steven Bird, E. Klein, and E. Loper. 2009. *Natural Language Processing with Python*. O'Reilly.

[Cor2010] IEL-UNICAMP and IME-USP, 2010. *Córpus Histórico do Português Anotado Tycho Brahe*.

[de Castilho1998] Ataliba Teixeira de Castilho. 1998. *Para a história do português brasileiro*, volume Vol I: Primeiras idéias. Humanitas, São Paulo.

[Kato and Roberts1993] Mary A. Kato and Ian Roberts. 1993. *Português brasileiro: uma viagem Diacrônica*. Editora da Unicamp, Campinas.

[Mattos e Silva1988] Rosa Virgínia Mattos e Silva. 1988. Fluxo e refluxo: uma retrospectiva da lingüística histórica no brasil. *D.E.L.T.A.*, 4(1):85–113.

[Megale and Cambraia1999] Heitor Megale and César Cambraia. 1999. Filologia portuguesa no brasil. *D.E.L.T.A.*, 15(1:22).

[Paixão de Sousa2004] Maria Clara Paixão de Sousa. 2004. Memórias do texto: Aspectos tecnológicos na construção de um corpus histórico do português. Projeto de pós-doutorado – fapesp, Unicamp.

[Paixão de Sousa2009] Maria Clara Paixão de Sousa. 2009. Desafios do processamento de textos antigos: primeiros experimentos na brasiliana digital. In *I Workshop de Linguística Computacional da USP*, São Paulo, 11.

[W3C1999] W3C. 1999. Extensible stylesheet language transformation.

[W3C2009] W3C. 2009. Extensible markup language.