AnCoraPipe: A new tool for corpora annotation

Working paper 1: TEXT-MESS 2.0 (Text-Knowledge 2.0)

Manuel Bertran, Oriol Borrega, M.Antònia Martí, Mariona Taulé, 2010
AnCoraPipe: A new tool for corpora annotation

Manuel Bertran*, Oriol Borrega**, M. Antònia Martí**, Mariona Taulé**
Centre de Llenguatge i Computació (CLiC)
University of Barcelona
Gran Via, 585, 08007 Barcelona.
*mbertran@lsi.upc.edu
**{oriol.borrega, amarti, mtaule}@ub.edu

Abstract
This paper describes AnCoraPipe, an environment for the creation, edition and analysis of linguistic corpora and lexicons. AnCoraPipe has been used in the development of different linguistic resources: AnCor, CesCa, ClInt, Amazighe corpora, and the verbal and nominal AnCora lexicons. We present the functionalities of AnCoraPipe, the way in which the data and metadata is structure, as well as some implementation details.

1. Introduction

AnCoraPipe is a computer environment for the creation, edition and analysis of linguistic corpora and lexicons. The efficiency and simplicity of the tools it incorporates was prioritized.

The motivation of the tool was (i) the need to develop large scale corpora (ii) with a wide variety of annotation levels, which (iii) implies several annotators working simultaneously and (iv) where version control is required. A friendly, user-oriented interface was created in order to ease annotator's work.

AnCoraPipe was implemented as a plugin in the Eclipse development platform, integrating all of its own tools and those available to the platform. Eclipse plugins facilitate the integrated management and collaborative development of corpora and lexicons.

Metainformation associated with the corpora and the lexicons is expressed by two means: (a) in terms of nodes and relations between nodes to mark document structure and to represent syntactic relations between constituents, and (b) in terms of attribute-value(s) associated to nodes for representing the rest of the information.

AnCoraPipe was designed considering two fundamental requirements: (1) Extensibility, that is, (a) the possibility to configure and modulate the set of attributes and values, making easier for the user the inclusion or exclusion of different levels of linguistic analysis, (b) the implementation of specialized annotation and view boards, and (c) the
adaptation of external tools for specific processing. (2) Multi-alphabet management: tools integrated in AnCoraPipe may be configured for working with any kind of alphabet. Up to now, it has been used for the treatment of corpora in Amazigh, Latin and Cyrillic alphabets.

The rest of the paper is organized as follows. In section 2 functionalities of AnCoraPipe are presented. Section 3 is devoted to the data and metadata structure. Section 4 includes the implementation details. In section 5 AnCoraPipe plugins are presented. In section 6 some extensions of the tool are described. Section 7 defines the basic working cycle. Finally section 8 and section 9 are devoted to applications and further work respectively.

2. Functionalities

AnCoraPipe carries out three main functionality sets: (a) the creation of new resources, (b) their edition and (c) the exportation or importation of data to or from other processing environments.

The creation of new resources can be done by the importation of textual data from external formats or by the creation of new documents within the platform itself.

Edition allows for the annotation of corpora and lexicons, and for the modification of previously annotated ones. The edition process is supported by graphical interfaces (windows) specific to each level of linguistic analysis. At present, the annotated levels include morphological, syntactic, semantic (lexical, sentential and Named Entities) and pragmatic (coreference and polarity) information.

Finally, AnCoraPipe supplies for the exportation of data for analysis using specialized tools such as Excel, SPSS, Weka, etc.

A subset of exportation tools enables the translation of AnCoraPipe format into other generic formats for the treatment and analysis of corpora. For instance, TBF format for the representation of syntactic trees, and the column format for the representation of dependency structures in the CoNLL² and SemEval³ competitions.

AnCoraPipe can be adapted to use external resources for automated or manual annotation. For example, it allows us to connect to a EuroWordNet database for the manual semantic annotation of corpora, and to reach different morphological analyzers available at FreeLing⁴ for the automatic annotation of Catalan, Spanish and English files.

---

¹ Small degree of programming may be useful for specialized tasks.
² http://www.cnts.ua.ac.be/conll2009/
³ http://stel.ub.edu/semeval2010-coref/
⁴ http://wwwlsi.upc.edu/~nlp/freeling/
3. Data and metadata structure

AnCoraPipe documents are XML documents with UTF-8 encoding. Other encodings are accepted as input, but the output of the platform will always be in UTF-8 format. Corpora and lexicons are stored in directories or folders containing the documents, texts -in the case of corpora- or lexical entries -in the case of lexicons. AnCoraPipe expects each file to contain a single document in order to simplify and facilitate the management of the data.

XML language was chosen as representation language because it is a standard that allows for representing any kind of data and supports any kind of encoding. UTF-8, in turn, supports the representation of texts in almost all existing writing systems. Tree-based XML document structure enables both the representation of the formal structure of documents as well as the syntactic structures of syntactically annotated corpora.

Nodes are the basic units of representation in XML. They are organized in a tree format where each node can have associated attribute-value pairs. Next, we present the typology of nodes and attribute-value pairs used in AnCoraPipe documents.

3.1. Nodes

Four different XML node types were defined to characterize textual documents. Each node contains zero or more attributes representing the information associated to it.

The nodes <article> and <sentence> are compulsive, constituting the higher nodes in the hierarchical structure. The node <sentence> has no restrictions regarding its structure and attributes.

<article>: Root node for all texts.
<sentence>: Associated to each sentence in the text.
<constituent>: In texts containing syntactic information, this node represents a generic constituent, which can take different 'names' according to the annotation carried out, such as: <sn>, <sp>, <sa>, etc. ¹
<word>: Generic node for words which takes the name of its morphological category (<n>, <v>, <d>, etc.) according to widely accepted standards (Eagles, 1996). In the current version, words are terminal nodes. Nevertheless, they could be treated as non-terminals if the representation of morphemes or phonemes is needed.

Lexical entries are also represented in XML document format. Lexicons have a fixed structure, constituted by the following nodes:

<lexentry>: Root node for each lexical entry.
<sense>: Contains all the frames of a given sense. Each lexical entry has

¹ <sn> stands for nominal phrase, <sp> for prepositional phrase, <sa> for adjectival phrase, etc.
from 1 to \( n \) senses.

*<frame>*: Specifies the argument structure of a sense. Each sense has from 1 to \( n \) frames.

*<argument>*: Describes the data of a frame argument. Each frame has from one to \( n \) arguments.

*<reference_modifier>*: Contains the constituents, being argumental or not. It is an optional node within the frame.

*<specifiers>*: Includes constituents which may be specifiers of the lemma. It is an optional node within the frame.

*<examples>*: Examples of the frame. It is an optional node within the frame.

### 3.2. Attributes

Linguistic information in AnCoraPipe documents is represented by attributes associated to the nodes -syntactic constituents or words- corresponding to each level of annotation: Morphological and lexical semantic\(^6\) information is associated to the node *<word>*. Syntactic information (constituents and functions), argument structure and thematic roles are assigned to nodes of the type *<constituent>*. Coreference is associated to both *<word>* and *<constituent>* nodes, and polarity can be assigned to *<word>* and *<sentence>* nodes.

Attributes were designed following the guidelines of databases normalization theory in order to avoid duplicates and make maintenance easier. The definition of attributes and their values is completely open and adaptable to all kinds of corpora and linguistic information.

All attributes and values as well as relations and restrictions between them are declared in XML tagset files. Concretely, in the tagsets are defined:

- Possible relations between nodes, i.e., which hierarchies are allowed.
- Which nodes may be terminal and which may not.
- The attributes and node names associated to one specific language.
- The mandatory requirements for attribute-value assignments.
- The attribute-value assignments triggered by a particular value.
- The attributes with an open list of values.
- The dynamic calculus of all possible values for an attribute using Java classes.

This open organization of data makes it easier to adapt the tool for the description of a wide variety of languages and to represent all kind of linguistic information.

As a drawback, it prevents from creating a DTD or XSD useful for all possible AnCoraPipe documents. We prioritized the expressivity of the format, leaving information robustness aside. To improve robustness, documents can be validated periodically by contrasting them with the tagsets

\(^6\) Named entities and WordNet senses.
or checking them whenever they are modified.

4. Implementation details

AnCoraPipe was implemented as a plugin for Eclipse, taking into account its integration with the generic tools of the platform. Eclipse is a free open coded development platform based in plugins, and it is available for major hardware architectures and operating systems. It has an active community of users who take care of maintenance and extensions.

AnCoraPipe can be integrated with other components. This is specially useful with the plugins which aim at facilitating collaborative work, such as SVN and CVS, that make project management (Tasks, Mylyn, etc.) and corpora annotation easier.

Eclipse extensions were implemented in JAVA with the SWT graphic library. SWT is not a JAVA standard, but it is included in Eclipse and, therefore, it has implementations available for all architectures it is distributed in, including the most popular ones: Windows, Linux, MacOS (Figure 1).

5. AnCoraPipe Plugins

AnCoraPipe can be used with any type of Eclipse distribution, although we recommend “Eclipse IDE for Java Developers”, which includes everything necessary for the installation.

AncoraPipe plugins distributions are publicly available at [http://clic.ub.edu/mbertran/ancorapipe/update](http://clic.ub.edu/mbertran/ancorapipe/update). Eclipse for RCP and RAP Developers might be useful for users aiming to extend the platform with new tools and Eclipse IDE for Java and Report Developers could be best for users aiming to generate some
Eclipse has two different graphic window types: Editor, which shows the whole content of a document, and View, which displays a partial, summarized or detailed view of a document or a specific aspect of the development environment. The configuration of these boards in a Eclipse's working window is called a Perspective.

5.1. Perspectives

Perspectives are graphical configurations of a group of graphic windows to carry out a particular task. AnCoraPipe plugins offer generic preset perspectives both for the treatment of corpora and for working in specific tasks. These perspectives can be modified and adapted by the user and saved for a later use.

Current perspectives are oriented to annotate morphology, argument structure, syntactic functions, Named Entities, WordNet senses, coreference, as well as to create and update lexicons.

5.2. Graphic search tool

AnCoraPipe has a generic search tool that uses XPath expressions (patterns) and evaluates them as groups of nodes, the set of nodes to be found. Searches can run over the whole workspace, over a few selected resources, over the projects the selected resources belong to or over resources in a working set.

Generic search has specific tools with preset patterns. For instance, it is possible to search for all nodes candidate to be Named Entity just asking for all noun phrases whose head is a proper noun without the NE attribute. Another search example is to ask for all the nodes lacking the argument and thematic role associated to its syntactic function.

The editor and the search utilities associated to it constitute a powerful tool to ensure the control and consistency of annotated data.

5.3. Document editor

The editor is the main component of AnCoraPipe, for it is used to edit documents from the corpus. The editor allows us to annotate new documents or modify previously existing ones. It is a generic tool that can be adapted according to the task to be done. It has three different views: structure, text and associated data, as it is shown in figure 2.

---

8 For more information about window sets, consult Eclipse help.
In the left column of figure 2, document structure is shown. On the top, the sentences of the document are listed and, in the bottom, we have the tree structure corresponding to the selected sentence. Tree structure is displayed in four columns: the leftmost one contains the node names, the second, relevant data for the process to be carried out, the third one shows the lemma of the word nodes, and in the rightmost one we find the words contained into each node.

The central column in figure 2 corresponds to text visualization and contains two windows. The upper one displays the text in the document separated in paragraphs. Any fragment of the text can be selected and AnCoraPipe will search the lowest node in the tree containing it. The structure corresponding to that node is shown in the lower part of the screen.

Finally, the right column corresponds to data visualization. The upper part allows for editing general document data such as language, source, author, etc. In the lower part, data to be shown can be selected.

The editor has a fast document search associated, which allows for looking for nodes in a document according to a variety of parameters. The search may proceed onwards or backwards from a selected node. Search criteria available are: by lemma, by word, by node path, by node containing a given attribute, by nodes containing a given attribute with a specific value, and by nodes matching Xpath patterns. These patterns are generated by selecting a criterion and fulfilling its possible values.

5.4. Views

Views are part of a perspective (see section 5.1). Even though views are part of preset perspectives, they can be dragged through the graphic space, opened and closed independently. Views allow for the visualization, edition and annotation of corpora and lexicons. For instance, constituents view can be used to display syntactic functions, thematic roles and arguments (figure 3).
Entities view (figure 4) shows all entities in a document with their constitutive mentions in the coreference chain. It is also possible to navigate through them.

The specialized views accelerate the whole annotation process. Each
level of annotation has its specialized view.

5.5. Lexical entry editor

The lexical entry editor facilitates the creation and updating of lexicons (figure 5). Lexical entries are organized in a fixed tree structure following standard XML format. The first level of the tree shows all senses in the entry, with their corresponding identifiers and an attribute showing whether they are lexicalized; for each sense, all frames are displayed with their corresponding types and argument structure. When a frame node is unfolded, all the information associated to it is shown in detail: syntactic function, argument structure, thematic roles, specifiers (in nominal frames), examples of usage, etc.

![Figure 5: Display of a verbal lexical entry](image)

5.6. Corpora import and export tools

One of the main advantages of AnCoraPipe is the capability to import and export data. AnCoraPipe has general tools to import documents in plain text and documents morphologically annotated with specific tools (for instance, FreeLing) and to convert them into AnCora XML format.

AnCoraPipe exporting tools are used (i) to transform AnCora format
in another one and (ii) to extract data from the corpus for specific uses. Up to date, three formats may be chosen:
a) Morphological file: the format obtained by analyzing a file with FreeLing.
b) The TBF format based on PennTreeBank (Marcus et al., 1993; Marcus et al., 1994).
c) The column format defined in ConLL 2008\(^*\) conference.

6. Extensibility

AnCoraPipe offers a variety of extension possibilities to create new tagsets (attributes and values), specialized views\(^{10}\) and perspectives, allowing users to quickly define new interface configurations to carry out new annotation levels.

AnCoraPipe extensibility is based in the “extension points” of Eclipse. Currently, AnCoraPipe includes an automatic annotation tool for morphology.

7. Working cycle

Once corpora are available, the basic working cycle in AnCoraPipe is as follows:
1) Import the corpus in AnCora format (if it is in another format) or synchronize the local working copy with the server in case of working in a collaborative system or a version server.
2) Search for all the nodes to be processed or manually reviewed.
3) Assign the iteration condition to get the next node every time a node is marked as reviewed.
4) For each node, review its contents, annotate new attributes, etc.
5) Update remote copies of the corpus if working with a collaborative system or a version server.

---


\(^{10}\) It is recommended to create specialized views for every new level of annotation included via tagsets.
8. Uses and applications

AnCoraPipe tools have been used in the development of the following resources:

- **AnCora**\(^{11}\) (Taulé et al., 2008; Recasens and Martí, 2010): two corpora of Spanish and Catalan (500kw each) annotated with morphology (lemma and PoS), syntax (constituents and functions), lexical semantics (WordNet senses and Named Entities), sentence semantics (arguments, thematic roles and semantic classes for verbs and nouns), coreference and polarity.

- **AnCoraVerb** (Aparicio et al., 2008) and **AnCoraNom lexicons**\(^{12}\) (Peris and Taulé, 2011): a verbal and a nominal lexicon for Catalan and Spanish.

- **AnCoraNet** (Taulé et al., 2011): a verbal multilingual lexicon (Catalan-Spanish-Basque) linked to the Unified Verb Index (UVI)\(^{13}\), including VerbNet, WordNet, FrameNet and PropBank.

- **CesCa**\(^{14}\) (Tolchinsky et al., 2010): a written Catalan corpus of scholars with great variation in orthography and typography which demanded a hard manual intervention.

- **ClInt**\(^{15}\) (Vila et al., 2010): an oral corpus of Clinical Interviews morphologically annotated.

\(^{11}\) [http://clic.ub.edu/corpus/ancora](http://clic.ub.edu/corpus/ancora)

\(^{12}\) [http://clic.ub.edu/corpus/ancora-lexics](http://clic.ub.edu/corpus/ancora-lexics)

\(^{13}\) [http://verbs.colorado.edu/verb-index/](http://verbs.colorado.edu/verb-index/)

\(^{14}\) [http://clic.ub.edu/cesca/](http://clic.ub.edu/cesca/)

• Amazighe corpus (Outahajala et al., 2010): a corpus of Amazighe language morphologically annotated.

9. Future work

Modular structure and IDE Eclipse integration make it possible to quickly widen AnCora tools. The expected extensions for the near future are:

• Possibility to use the fields of a database as source documents.
• Implementation of standard extension points for AnCoraPipe plugins.
• Integration of automatic annotators as AnCoraPipe tools.
• Implementation of contextual menus.
• Implementation of a periodical checking system for corpora.

Acknowledgements

This work was supported by the projects Text-Knowledge 2.0 (TIN2009-13391-C04-04), AnCora-Net (FFI2009-06497-E/FILO) and ClInt (FFI2009-06252-E/FILO) from the Spanish Ministry of Science and Innovation.

References


