Implementing Greek Morphology

Helma Dik
University of Chicago
helmadik@mac.com

Richard Whaling
University of Chicago
rwhaling@uchicago.edu

In this poster we discuss the nuts and bolts of our implementation of Greek morphology in a five-million word corpus, that of the Perseus Greek texts. Many disparate elements, and the efforts of many different people have come together in this project. Dik & Whaling (2008) describe how initial data was gathered from multiple sources; the current paper describes what went into the final product:

Disambiguated Greek texts: Early Greek Epic and the New Testament

The two sources of data from which we were going to bootstrap our project came with their own specific features: The two disambiguated corpora used different data specifications, which had to be compared and made uniform, and brought up to the standard that we wanted. However, it did bring us two large swaths of data of 350K words in total with which to seed our part-of-speech analyser, TreeTagger.

Morphological analysis from the Perseus project

The training data alone were not going to be adequate to produce a full lexicon for TreeTagger. We decided to supplement it with the output from Perseus's Morpheus tool (Crane 1991) of all possible parses for the full corpus. This greatly enhanced TreeTagger's accuracy on rare words not encountered in the training data, but also generated many redundancies and inconsistencies which made it hard for TreeTagger to build a proper decision tree. It was a continuing dilemma to those involved in the project whether we should allow more correct input to eventually weed out the many incorrect parses, or to remove incorrect parses from the input directly. Some effort was made to remove incorrect input: A 28 MB lexicon was reduced to less than 23 MB, but this represented only a portion of problematic entries.

TreeTagger

TreeTagger (Schmid 1995) is the proprietary software we used to assign part-of-speech tags (in effect, a full morphological disambiguation in a ten-slot morphological code plus a lemma). We trained TreeTagger in the first instance on the basis of the New Testament data, and added 40,000 words total in representative 1000 word samples from the rest of the corpus. New sets of samples were prepared with TreeTagger disambiguation, for which we used earlier disambiguated samples, plus our initial New Testament samples, as input.

Disambiguation (internal)

On the basis of earlier work on Czech and other languages, we decided that 40,000 words would be an adequate sample. This disregarded the fact that most research in natural language processing is actually done on more homogeneous texts than our samples of Greek literature, such as Reuters news items. Clearly, a more homogeneous input makes for higher accuracy within the source corpus, but we had no such luxury. An early indication was the high accuracy rate achieved by TreeTagger when trained and tested on Homeric Greek, which is a highly homogeneous, formulaic, subset of our texts. Perhaps the Homeric corpus is in fact the best parallel to Reuters and similar corpora in modern languages - at least the accuracy was comparable.

In more practical terms, undergraduate students of Greek were hired to 'pick the right parse' from among possible parses identified by TreeTagger. The disambiguation interface allowed the students to signal alternative parses or lemmas if none of the TreeTagger choices was accurate. Next, in an 'admin' layer, items about which there were disagreements among the students or about which comments were entered, were highlighted for review, so that the principal investigator could review these items especially, prior to feeding fully disambiguated texts back to TreeTagger.

Implementation

The centerpiece of our implementation is a SQLite database backend, containing the tokens and parses for the full corpus. It connects the three major components of the system:

- The original Perseus XML files, in which the tokens have been given unique ids as follows, keeping intact all previous markup:

  <w id="276565">ὦ</w>

  <w id="276566">ἄνδρες</w>

  <w id="276567">Ἀθηναῖοι</w

- TreeTagger, which accepts token sequences from the database and outputs parses and probability weights, which are stored in their own table.

- PhiloLogic, which serves as a highly efficient search and retrieval front end, by indexing the augmented XML files as well as the contents of the linked SQLite tables. PhiloLogic's highly optimized index architecture allows near-instantaneous results on complex inquiries such as 'any infinitive forms within 25 words of (dative singulars of) lemma X and string Y', which would be a challenge for typical relational database systems.

For a concrete example, in a standard PhiloLogic search box, entering 'lemma:μῆνις' will produce this word from the first line of the Iliad, as will a search for 'pos:*fa*', as will a search for the original string, 'μῆνιν'. Criteria can be combined as well, so that 'lemma:μῆνις;pos:*fa*' produces only feminine accusative forms of the particular lemma μῆνις.

We continue to explore the possibility of natural language searching as a substitute or alternative to this highly technical way of querying the corpus, and will demonstrate our progress on this front at the conference. The goal is to make it possible for users to type 'feminine accusative' as opposed to 'pos:*fa*', which will remain daunting to all but the most determined.

Conclusion

We are happy to have disambiguated a large corpus, making available for the first time a large, representative corpus of Classical Greek for morphological searching in addition to searching by string and by lemma — integrated into the existing reading and browsing environment for the texts. However, we are now also prepared to start crowd-sourcing the long tail of incorrect parses. Besides looking up the statistically most probable parse according to TreeTagger and other possible parses, users can 'vote' to correct TreeTagger's chosen parses. Once these votes have been inspected and accepted into the main database, future updates to the corpus will reflect both these local corrections and, over the full extent of the corpus, a more accurate TreeTagger. It is our hope that with the assistance of our users we will approach higher and higher levels of accuracy, making this tool ever more useful to scholars of Classical Greek.

References:

Crane, Gregory (1991). Generating and parsing classical Greek. In *Literary and Linguistic Computing*, 6(4): 243-245, 1991.

Dik, Helma and Richard Whaling (2008) - Bootstrapping Greek Morphology. Digital Humanities 2008.

Schmid, Helmut (1995) - Improvements in part-of-speech tagging with an application to German. In Proceedings of the ACL SIGDAT-Workshop. http://www.ims.uni-stuttgart.de/ftp/pub/corpora/tree-tagger2.pdf

Website URL: http://perseus.uchicago.edu