Bootstrapping Classical Greek Morphology

Helma Dik
helmadik@mac.com
University of Chicago, USA

Richard Whaling
rwhaling@uchicago.edu
University of Chicago, USA

In this paper we report on an incremental approach to automated tagging of Greek morphology using a range of already existing tools and data. We describe how we engineered a system that combines the many freely available resources into a useful whole for the purpose of building a searchable database of morphologically tagged classical Greek.

The current state of the art in electronic tools for classical Greek morphology is represented by Morpheus, the morphological analyzer developed by Gregory Crane (Crane 1991). It provides all possible parses for a given surface form, and the lemmas from which these are derived. The rich morphology of Greek, however, results in multiple parses for more than 50% of the words (http://grade-devel.uchicago.edu/morphstats.html). There are fully tagged corpora available for pre-classical Greek (early Greek epic, developed for the 'Chicago Homer', http://www.library.northwestern.edu/homer/) and for New Testament Greek, but not for the classical period.

Disambiguating more than half of our 3 million-word corpus by hand is not feasible, so we turned to other methods. The central element of our approach has been Helmut Schmid's TreeTagger (Schmid 1994, 1995). TreeTagger is a Markov-model based morphological tagger that has been successfully applied to a wide variety of languages. Given training data of 20,000 words and a lexicon of 350,000 words, TreeTagger achieved accuracy on a German news corpus of 97.5%. When TreeTagger encounters a form, it will look it up in three places: first, it has a lexicon of known forms and their tags. Second, it builds from that lexicon a suffix and prefix lexicon that attempts to serve as a a morphology of the language, so as to parse unknown words. In the absence of a known form or recognizable suffix or prefix, or when there are multiple ambiguous parses, it will estimate the tag probabilistically, based on the tags of the previous *n* (typically two) words; this stochastic model of syntax is stored as a decision tree extracted from the tagged training data.

Since classical Greek presents, prima facie, more of a challenge than German, given that it has a richer morphology, and is a non-projective language with a complex syntax, we were initially unsure whether a Markov model would be capable of performing on Greek to any degree of accuracy. A particular complicating factor for Greek is the very large tagset: our full lexicon contains more than 1,400 tags, making it difficult for TreeTagger to build a decision tree from small datasets. Czech

(Hajič 1998) is comparable in the number of tags, but has lower rates of non-projectivity (compare Bamman and Crane 2006:72 on Latin).

Thus, for a first experiment, we built a comprehensive lexicon consisting of all surface forms occurring in Homer and Hesiod annotated with the parses occurring in the hand-disambiguated corpus--a subset of all grammatically possible parses--so that TreeTagger only had about 740 different possible tags to consider. Given this comprehensive lexicon and the Iliad and Odyssey as training data (200,000 words), we achieved 96.6% accuracy for Hesiod and the Homeric Hymns (see http://grade-devel.uchicago.edu/tagging.html).

The experiment established that a trigram Markov model was in fact capable of modeling Greek grammar remarkably well. The good results can be attributed in part to the formulaic nature of epic poetry and the large size of the training data, but they established the excellent potential of TreeTagger for Greek. This high degree of accuracy compares well with state-of-the-art taggers for such disparate languages as Arabic, Korean, and Czech (Smith et al., 2005).

Unfortunately, the Homeric data form a corpus that is of little use for classical Greek. In order to start analyzing classical Greek, we therefore used a hand-tagged Greek New Testament as our training data (160,000 words). New Testament Greek postdates the classical period by some four hundred years, and, not surprisingly, our initial accuracy on a 2,000 word sample of Lysias (4th century BCE oratory) was only 84% for morphological tagging, and performance on lemmas was weak. Computational linguists are familiar with the statistic that turning to out-of-domain data results in a ten percent loss of accuracy, so this result was not entirely unexpected.

At this point one could have decided to hand-tag an appropriate classical corpus and discard the out-of-domain data. Instead, we decided to integrate the output of Morpheus, thereby drastically raising the number of recognized forms and possible parses. While we had found that Morpheus alone produced too many ambiguous results to be practical as a parser, as a lexical resource for TreeTagger it is exemplary. TreeTagger's accuracy on the Lysias sample rose to 88%, with much improved recognition of lemmas. Certain common Attic constructs, unfortunately, were missed wholesale, but the decision tree from the New Testament demonstrated a grasp of the fundamentals.

While we are also working on improving accuracy by further refining the tagging system, so far we have seen the most prospects for improvement in augmenting our New Testament data with samples from classical Greek: When trained on our Lysias sample alone, TreeTagger performed at 96.8% accuracy when tested on that same text, but only performed at 88% on a new sample. In other words, 2,000 words of in-domain data performed no better or worse than 150,000 words of Biblical Greek combined with the Morpheus lexicon. We next used a

combined training set of the tagged New Testament and the hand-tagged Lysias sample. In this case, the TreeTagger was capable of augmenting the basic decision tree it had already extracted from the NT alone with Attic-specific constructions. Ironically, this system only performed at 96.2% when turned back on the training data, but achieved 91% accuracy on the new sample (http://grade-devel.uchicago.edu/Lys2.html for results on the second sample). This is a substantial improvement given the addition of only 2,000 words of text, or less than 2% of the total training corpus. In the longer term, we aim at hand-disambiguating 40,000 words, double that of Schmid (1995), but comparable to Smith et al. (2005).

We conclude that automated tagging of classical Greek to a high level of accuracy can be achieved with quite limited human effort toward hand-disambiguation of in-domain data, thanks to the possibility of combining existing morphological data and machine learning, which together bootstrap a highly accurate morphological analysis. In our presentation we will report on our various approaches to improving these results still further, such as using a 6th order Markov model, enhancing the grammatical specificity of the tagset, and the results of several more iterations of our bootstrap procedure.

References

Bamman, David, and Gregory Crane (2006). The design and use of a Latin dependency treebank. In J. Hajič and J. Nivre (eds.), *Proceedings of the Fifth International Workshop on Treebanks and Linguistic Theories* (TLT) 2006, pp. 67-78. http://ufal.mff.cuni.cz/tlt2006/pdf/110.pdf

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

Hajič, Jan (1998). Building a syntactically annotated corpus: The Prague Dependency Treebank. In Eva Hajičova (ed.), *Issues of Valency and Meaning,* pp. 106-132.

Schmid, Helmut (1994). Probabilistic part-of-speech tagging using decision trees. In *International Conference on New Methods in Language Processing,* pp. 44-49. http://www.ims.uni-stuttgart.de/ftp/pub/corpora/tree-tagger1.pdf

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

Smith, Noah A., David A. Smith, and Roy W. Tromble (2005). Context-based morphological disambiguation with random fields. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing,* pp. 475-482. http://www.cs.jhu.edu/~dasmith/sst_emnlp_2005.pdf