Evaluating automatically annotated treebanks for linguistic research

1. Introduction

Automatically annotated linguistic resources contain the 'big data' of linguistics, but in some cases they contain bad data as well. In this work, I propose and compare four different approaches to distinguishing big data from bad data when a particular linguistic phenomenon is being studied. Current methods of evaluation are too general for the purposes of studying a particular phenomenon.

Treebanks are text corpora that have been enriched with syntax trees, allowing linguists to search the texts for particular syntactic constructions and properties. Such searching will result in a list of only those constructions, which is much easier to study than an entire text. Due to advances in natural language processing, it has become possible to syntactically parse large amounts of text automatically, with fairly good accuracy. This has resulted in the creation of treebanks that are much larger than traditional, manually annotated ones. The main advantage of these large-scale treebanks is that they can be used to investigate rare constructions, co-occurrence patterns of uncommon words, or small probabilistic effects. They also provide larger sample sizes for more common phenomena. The main disadvantage is the error rate. While manually annotated or manually checked treebanks contain some errors, automatic annotation comes at the cost of annotation accuracy. The errors made by the parser will include systematic errors, where the parser was unable to annotate a particular construction correctly, thereby failing to annotate any instance of it correctly. Therefore, when using automatically annotated treebanks for linguistic study, some sort of evaluation is necessary to make sure that the construction of interest was annotated correctly (most of the time).

2. Applications

Automatically annotated treebanks are a useful source of information in any study where large sample sizes are beneficial. Such treebanks have been made available for various languages. For Dutch, there is the 700 million word Lassy Large treebank (van Noord et al., 2013). For German, the 200 million word TüPP-D/Z (Müller, 2004) is available with automatic annotation. For English, the Google Books n-gram corpus (Lin et al., 2012) has been annotated syntactically, as well as the 4 billion word Gigaword v5 corpus (Napoles et al., 2012). There are other examples, and treebanks for a specific domain or language can be created as long as an automatic parser is available. This is the case for many major languages.

These treebanks have already been used to study various linguistic phenomena. For Dutch, several applications of the Lassy Large treebank are illustrated in van Noord and Bouma (2009). A study of extraposition of comparative objects by van der Beek et al. (2002) was used to illustrate the grammar used by the Alpino parser, but it attracted criticism for allowing too much extraposition. A reviewer claimed that such extraposition was not possible from the front of the sentence (the topic), however, a search of the large corpus revealed that such sentences were in fact being used in particular contexts. It was judged to be a probabilistc phenomenon, more or less acceptable depending on various factors. Bastiaanse and Bouma (2007) used syntactic structures from the treebank to argue that patients with Broca's aphasia have difficulty with constructions of higher linguistic complexity, rather than due to a frequency phenomenon. Bouma and Spenader (2008) studied the distribution of the Dutch reflexives *zich* and *zichzelf* with regards to the verbs they occur with, where different verbs can select one or both of the options.

Bloem et al. (2014) used a part of the Lassy Large treebank to study Dutch verb cluster constructions, a word order variation in which many factors play a probabilistic role. This involves searching the treebank for groups of verbs in a particular syntactic configuration: an auxiliary or modal verb heading a participal or infinitival main verb. This study also replicates earlier work on a manually annotated corpus, showing that the errors caused by the automatic parsing are not necessarily a problem for linguistic study, although a few factors (i.e. word stress patterns) could not be studied due to the nature of the annotation that can be found in a treebank of written texts.

Using the TüPP-D/Z treebank, auxiliary fronting in German was studied (Hinrichs and Beck, 2013). Since this is a relatively rare construction, the massive size of the corpus was a requirement to be able to find enough instances. The authors observe what verbs participate in the construction, and compare the treebank data to (much more sparse) information from diachronic corpora. For English, Lehmann and Schneider (2012) used a 580 million word dependency-parsed corpus to study the influence of specific lexical types on the English dative alternation. These types consist of 'triplets' of words: a ditransitive verb, a direct object head and an indirect object head — these slots are all filled with open-class words, requiring massive amounts of data to study.

3. Current approaches to evaluation

The quality of automatically annotated corpora is usually evaluated by testing the performance of the parser that was used to create it. Therefore, they are evaluated in the same way as parsers, using an overall accuracy score such as the word-based Attachment Score. This is the percentage of words that have been assigned the correct head in the syntactic structure (sentence-based variations or variations that include dependency labeling also exist). The Lassy Large

corpus was evaluated using Concept Accuracy (the proportion of correct labeled dependencies) as a measure. A part of the corpus containing texts from various domains (e.g. books, newspaper texts) was manually verified in order to have a gold standard to compare against. This resulted in an accuracy score of 86.52%, but with clear variation across different domains (van Noord, 2009).

However, even this is too general for the purposes of linguistic research. Rather than some domain of text, a researcher is generally interested in one particular construction, and wants to know how accurately that particular construction was parsed in the corpus. If the parser often errs in labeling adjectives, this does not matter if one wants to investigate reflexives, but it would be a major problem for a study of adjectives. Therefore, I propose that semi-automatic or manual construction-specific evaluation is necessary, using the knowledge of linguistic experts. In the next section I will discuss four possible approaches, illustrating them with examples from the Dutch verb cluster research described by Bloem et al. (2014)

4. Linguistically informed evaluation

The most obvious alternative is a complete **manual evaluation of the results** by a linguist, but this has its downsides. Firstly, it may take a lot of time and resources to evaluate all results from a large corpus in this way. And secondly, this method may still miss constructions that were systematically misparsed. For example, if a researcher is searching for verb clusters but some verbs have been mistagged as adjectives, a search query for verb clusters will not find those mistagged instances, and the researcher will not know of their existence. The precision of the results (the percentage of found instances that are relevant) can be measured with such an evaluation, but not the recall (the percentage of relevant instances that are found).

To counter this, it may be possible to do a **manual evaluation of the text**. By reading the original corpus text rather than just the results of a search query, even instances of the construction of interest that are completely misparsed can be found by the linguist. However, this is extremely labour-intensive — one will have to read a lot of text to find just one instance of a rare construction, even if only a part of the corpus is evaluated in this way. This takes away the main advantage of using a large automatically parsed treebank.

Another solution is to **fall back to simpler annotation**. It is generally the case that annotation of larger structures is more difficult. Lemmatizing and tagging (assigning a word class) only involves words, while parsing adds syntactic structure over multiple words. Queries based on word class only will therefore result in fewer errors than searching on the basis of syntactic structure. For example, to retrieve verb cluster construction one would normally want to find a verb that is the head of another verb. But in this way, verbs that were attached incorrectly will erroneously be skipped. If the researcher simply searches for two verbs positioned next to each other in the linear order, these skipped verbs would also be included, at the cost of also retrieving verbs that are next to each other coincidentally (i.e. as part of two different clauses). Comparing the result of the two procedures will produce a list of 'suspicious' instances, which can be evaluated manually (to be either included or excluded from the study) with less effort and better recall than when the results of a regular corpus query are manually evaluated. This does mean that the linguist will have to come up with some sort of word-class-based approximation of the construction under study using their knowledge of the language.

Lastly, it is possible to **search for particular instances** of the construction without relying on the annotation at all. The linguist can choose some representative instances of the construction and search for it directly. For example, when searching for Dutch verb clusters, one could simply search the corpus for the string 'hebben gezien' (have seen), one of many possible combinations of verbs. I will call this a 'string query', as opposed to a 'syntactic query' that one would normally perform on a treebank. This will only result in a limited number of results, but in a large automatically annotated corpus there can still be many results for a specific combination of words, even if the total number of instances of the general construction (verb-verb combinations in this case) is much larger. This string query does not rely on any syntactic annotation, and it would therefore find the verbs even if they were annotated completely erroneously, i.e. as a preposition heading a preposition. These results for the string 'hebben gezien' can then be compared to results for the syntactic structure of 'hebben gezien' with these particular words to see whether there is a recall problem: if an example occurs in the string query but not in the syntactic query, it is annotated incorrectly in a way that makes it impossible to find with a syntactic query. If the researcher does this for various instances of the construction, they should get a clear idea of the reliability of the annotation, and what sort of errors to look out for. However, it would be impossible to find all annotation errors in this way, since for most research questions it would be impossible to search for every possible instantiation of a construction.

5. Conclusion

In summary, there are various ways to evaluate linguistic data from automatically annotated treebanks that may help to alleviate the concerns that linguists often have about the inaccuracies of such corpora. Since the proposed methods all have different advantages and disadvantages, it would be best to combine them when studying a particular construction. **Manual evaluation of the results** can be used to determine the precision of a corpus query's results, while **searching for particular instances** can be used to calculate recall over a portion of the data, determining how many examples might have been missed. **Falling back to simpler annotation** can be used as a verification of the syntactic annotation of the corpus, even over larger amounts of data. These methods allow a linguistic researcher to get a good impression of the quality of the annotation of the particular construction they are investigating in the corpus, while still preserving the advantage of being able to obtain many examples with relatively little manual effort.

6. References

Bastiaanse, R. and Bouma, G. (2007). Frequency and linguistic complexity in agrammatic speech production. *Brain and Language*, 103(1):78–79.

Bloem, J., Versloot, A., and Weerman, F. (2014). Applying automatically parsed corpora to the study of language variation. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1974–1984, Dublin, Ireland, August. Dublin City University and Association for Computational Linguistics.

Bouma, G. and Spenader, J. (2008). The distribution of weak and strong object reflexives in Dutch. *LOT Occasional Series*, 12:103–114.

Hinrichs, E. and Beck, K. (2013). Auxiliary fronting in german: A walk in the woods. In *The Twelfth Workshop on Treebanks and Linguistic Theories (TLT12)*, page 61.

Lehmann, H. M. and Schneider, G. (2012). Syntactic variation and lexical preference in the dative-shift alternation. *Language and Computers*, 75(1):65–75.

Lin, Y., Michel, J.-B., Aiden, E. L., Orwant, J., Brockman, W., and Petrov, S. (2012). Syntactic annotations for the Google Books Ngram corpus. In *Proceedings of the ACL 2012 system demonstrations*, pages 169–174. Association for Computational Linguistics.

Müller, F. H. (2004). Stylebook for the tübingen partially parsed corpus of written german (tüpp-d/z). In *Sonderforschungsbereich 441, Seminar für Sprachwissenschaft, Universität Tübingen*, volume 28.

Napoles, C., Gormley, M., and Van Durme, B. (2012). Annotated gigaword. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction*, pages 95–100. Association for Computational Linguistics.

van der Beek, L., Bouma, G., and van Noord, G. (2002). Een brede computationele grammatica voor het nederlands. *Nederlandse Taalkunde*, 2002:353–374.

van Noord, G. and Bouma, G. (2009). Parsed corpora for linguistics. In *Proceedings of the EACL 2009 Workshop on the Interaction between Linguistics and Computational Linguistics: Virtuous, Vicious or Vacuous?*, pages 33–39. Association for Computational Linguistics.

van Noord, G., Bouma, G., Van Eynde, F., de Kok, D., Linde, J., Schuurman, I., Sang, E. T. K., and Vandeghinste, V. (2013). Large scale syntactic annotation of written Dutch: Lassy. In Spyns, P. and Odijk, J., editors, *Essential Speech and Language Technology for Dutch*, Theory and Applications of Natural Language Processing, pages 147–164. Springer Berlin Heidelberg.

van Noord, G. (2009). Huge parsed corpora in lassy. *Proceedings of TLT7. LOT, Groningen, The Netherlands*.