Open challenges in treebanking: some thoughts based on the Copenhagen Dependency Treebanks

Matthias Buch-Kromann
Copenhagen Business School

E-mail: matthias@buch-kromann.dk

Abstract

Despite the obvious importance and success of treebanks and other linguistically annotated corpora in the current data-driven statistical paradigm in computational linguistics, there are many outstanding challenges. This paper identifies some of the more important challenges, which are mainly concerned with how to exploit synergies by linking up different annotation projects with each other, how to link linguistic annotations to linguistic theory, and how to assess treebank quality without unintended distortions in the way research is conducted in the field.

1 Introduction

Since the creation of the Penn Treebank by Marcus et al [9] in the early 1990's, linguists and computational linguists have succeeded in creating a large number of excellent linguistically annotated corpora (or treebanks, for short). These treebanks cover a large number of languages and a wide range of different linguistic levels, most importantly syntax, part-of-speech, morphology, discourse, coreference, predicate-argument structure, semantics, and word and phrase alignments. They differ from dictionaries and other lexical resources in that they encode linguistic analyses of language phenomena in context, rather than linguistic analyses of words in isolation, and this is the key to their success.

The treebanks created by the field are an important achievement which, together with new statistical techniques, have fuelled the recent paradigm change in computational linguistics from rule-based systems whose language-specific knowledge is encoded as hand-made dictionaries and grammars to data-driven statistical systems whose language-specific knowledge is induced from raw and annotated texts. This paradigm change has spurred the development of a wide range of supervised and semi-supervised statistical techniques in natural language processing that build on annotated corpora — with statistical parsing as the most prominent and successful application so far. Although fully unsupervised techniques have been proposed in many areas of natural language processing, they have so far mostly failed to produce results that are competitive with supervised or semi-supervised methods, with machine translation as a remarkable exception. Since nothing suggests an imminent breakthrough in fully unsupervised methods outside MT, there is every reason to believe that annotated corpora will continue to play a crucial role as the primary source of language-specific knowledge in most statistical systems in the foreseeable future.

However, despite the obvious importance and success of treebanks in the new statistical paradigm, there are many outstanding challenges. Some of these concern how treebanks are created, some concern how they are utilized, and some how they are compared, merged, and evaluated. In the following, I will describe what I see as some of the main challenges facing the field, and outline some of the tentative steps that have been, or in my opinion should be, taken towards resolving them.

2 The main challenges

Challenge 1: Bridging between different annotation schemes

Current treebanks are based on a rich variety of linguistic theories and annotation schemes. A single, one-size-fits-all annotation scheme covering all languages and linguistic levels is probably neither desirable, nor possible. But the confusion that arises from the current wealth of annotation schemes is a major obstacle in the development of systems that build on treebanks. The problem is particularly acute for systems that need to draw on several different treebanks simultaneously. For example, Chiang [6] identifies badly interacting source and target language annotation schemes as an important obstacle in supervised tree-to-tree translation, and Meyers [13, 11] points to the lack of coordination between annotation projects as a major obstacle in annotation merging.

An example of the kind of confusion that may arise, even between closely related schemes, is illustrated by Figures 1 and 2, which show the syntactic analysis of four different constructions in two dependency-based annotation schemes: the dependency conversion of the Penn Treebank produced by the PennConverter [8], and the native dependency annotation used in the 100,000 word English part of the Copenhagen Dependency Treebanks [5, 3], the second-largest native (non-converted) dependency treebank for English after the Prague English Dependency Treebank [7]. Although the two annotation schemes are both dependency-based and therefore fall within the same broader family of annotation schemes, the specific differences between the analyses are considerable (non-shared unlabeled arcs are shown as dotted red arcs). The PennConverter scheme takes a semantics-oriented view, motivated by its intended use as training material for parsing semantic dependencies, where content words (nouns, main verbs) tend to dominate function words (determiners, auxiliary verbs, prepositions). In contrast, the CDT scheme takes a syntax-oriented view, where function words tend to dominate content words. There are other differences as well, eg, with respect to the analysis of

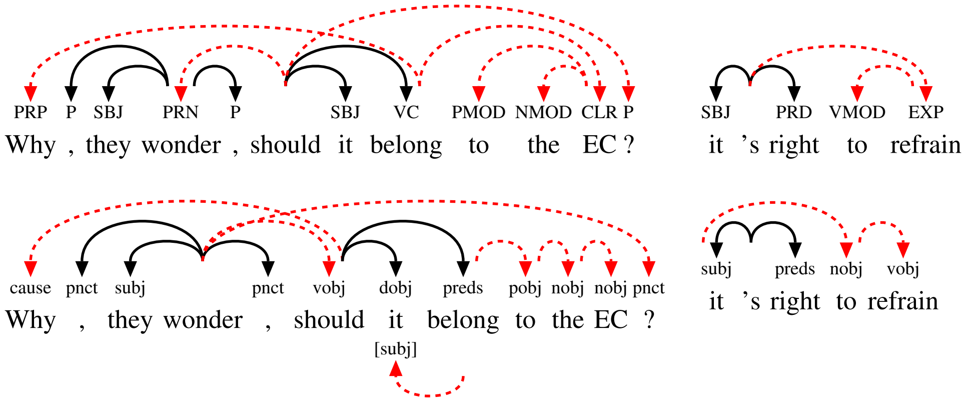
PRP P SBJ PRN P SBJ VC PMOD NMOD CLR P
Why , they wonder , should it belong to the EC ?
SBJ PRD VMOD EXP
it ’s right to refrain
cause pnct subj pnct vobj dobj preds pobj nobj nobj pnct
Why , they wonder , should it belong to the EC ?
[subj]
subj preds nobj vobj
it ’s right to refrain

Figure 1: Differences between PennConverter’s dependency conversion of the Penn Treebank (top) and the CDT scheme (bottom): attribution (left) and expletive construction (right). (Differing arcs shown with dashed lines.)

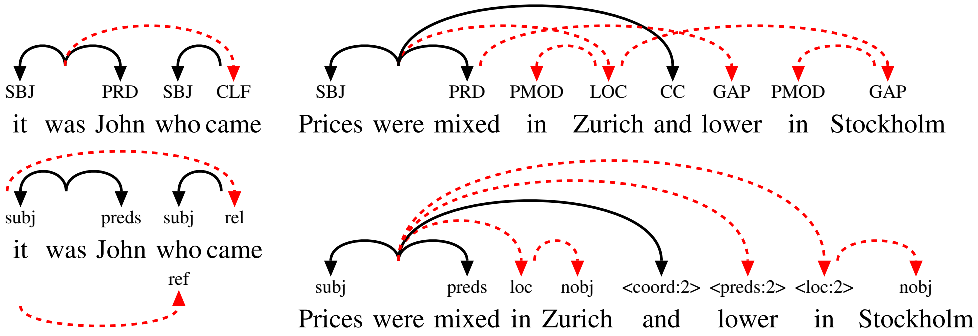
SBJ PRD SBJ CLF
it was John who came
SBJ PRD PMOD LOC CC GAP PMOD GAP
Prices were mixed in Zurich and lower in Stockholm
subj preds subj rel
it was John who came
ref
subj preds loc nobj <coord:2> <preds:2> <loc:2> nobj
Prices were mixed in Zurich and lower in Stockholm

Figure 2: Differences between the PennConverter scheme (top) and the CDT scheme (bottom): cleft sentence (left) and gapping (right)

attribution verbs, gapping, extrapositions, cleft sentences, and the use of additional secondary dependencies and coreference links in the CDT treebanks.

Both schemes are linguistically well-motivated, so it is not a question about one scheme being right and the other being wrong. Indeed, with differences like these, there is no objective criterion for deciding which annotation framework provides the most empirically adequate analysis of the texts. From a theoretical point of view, it may even be misleading to talk about the best scheme, as if there is only one: since we are modelling unobservable properties of language, there is no guarantee that we cannot end up with a set of highly different models which are equally adequate with respect to their observable consequences. For this reason, the relevant challenge is not to create a single unified annotation scheme to be used by all treebanks created in the field, an impractical and unrealistic task — the relevant challenge is to find better ways of translating between different annotation schemes and merging them, in order to pool costly treebank resources.

There are probably many ways of achieving this goal, but the diagram in Fig-

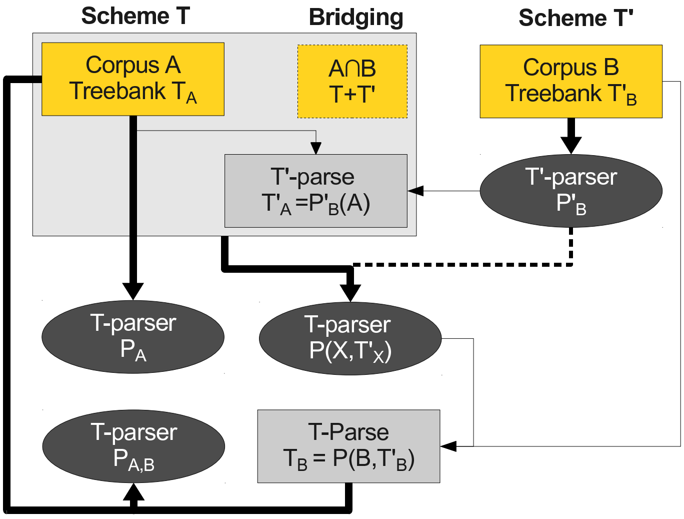

Figure 3: Treebank conversion: Converting a T'-treebank T'_B to a T-treebank T_B, and creating T-parsers $P(X,T'_X)$ and $P_{A,B}$ by pooling treebanks.

ure 4 illustrates the kind of system that I have in mind for treebank conversion and automatic annotation (for simplicity, I will refer to a linguistically annotated corpus T as a 'treebank', and to an automatic annotation system P trained on T as a 'parser'). In the diagram, I am assuming that we have two annotation schemes T,T' applied to two corpora A,B respectively, yielding manually corrected "gold" treebanks T_A,T'_B. For example, T_A might be the Copenhagen Dependency Treebank for English, T'_B might be the Penn Treebank, and my goal might be to convert the Penn Treebank to CDT format, or to build a better CDT parser that would use the Penn Treebank as training material.

We can create parsers P_A,P'_B without any pooling by training them on T_A,T'_B independently. But in order to pool the treebanks, we need to do something else. Inspired by stacked dependency parsing,[1] one possibility is to use the P'_B parser to create an automatically parsed T'-corpus T'_A, and use T_A,T'_A to create a stacked T-parser $P(X,T'_X)$ where T'_X is a T'-annotation of X, eg, a parse produced by P'_B. This stacked parser can then be used to convert T'_B into a T-treebank $T_B = P(B,T'_B)$. Finally, the original treebank T_A and the converted treebank T_B can be pooled to train a new T-parser $P_{A,B}$. The parser $P_{A,B}$ and the stacked parser $P(X,P'_B(X))$ will utilize the information from both T_A and T'_B, and can therefore be expected to perform better than the P_A parser, especially if T_A is a large high-quality treebank. When using the stacked parser $P(X,T'_X)$ as a conversion system, it would probably be helpful to include a designated *bridging treebank* in the system, ie, an overlapping subcorpus $A \cap B$ where the T' annotation has been hand-converted into the

[1] Stacked dependency parsing [16, 10, 18], which is the current state-of-the-art approach in dependency parsing, employs two or more dependency parsing systems to a single treebank, so our setup is a slight variation of the original setup.

corresponding T annotation.[2]

As a variation on this setup, the corpus B might consist of parallel texts, with A as a subcorpus of B, and T'_B, T_A as treebank annotations of the source and target texts in B and A, respectively; $T_A + T'_A$ will then be available as a bridging treebank. The goal is to extend the T-annotation to B. For example, in the CDT treebanks, T'_B might be the 100,000 word treebank for the source language Danish, and T_A might be one of the 30,000 word treebanks for the target languages English, German, Italian, or Spanish, and the goal is to extend the target language treebank to all translations of texts in the Danish treebank. In this case, we would like to train a stacked parser $P(X, T'_X)$ from $T'(A)$ and the bridging treebank $T + T'$ on A, and use the parser to produce $T_B = P(B, T'_B)$.

To sum up, the challenge is to make it easier to convert annotations from one scheme to another, and to create automatic annotation systems that can utilize multiple treebanks with different annotation schemes simultaneously. Solving this challenge will have great practical value, because it will make it easier to convert one treebank format into another, transfer treebank annotations from one language to another, and create monolingual and synchronous parsing systems that build on pooled treebank resources. Small, high-quality bridging treebanks can be expected to improve the quality of these systems, and building bridging treebanks and determining how large they need to be, is therefore an important task for future research.

Challenge 2: Bridging annotations at different linguistic levels

Many annotation projects take a narrow scope where they focus on a single language, a single linguistic level, and perhaps even a single text genre. Many of the most influential annotated corpora for English are based on a narrow-scope approach: eg, Penn Treebank, Penn Discourse Treebank, PropBank, NomBank, TimeBank, and the MATE/GNOME scheme for coreference annotation. Other annotation projects take a wide scope where they seek to provide a coherent unified annotation for a wide variety of languages, linguistic levels, and text genres: eg, the Prague Dependency Treebanks (Czech, English, Arabic), the Copenhagen Dependency Treebanks (Danish, English, German, Italian, Spanish), and OntoNotes (English), which cover syntax, morphology, semantics, discourse, and coreference.

From a scientific perspective, the narrow-scope and wide-scope approaches complement each other: narrow-scope encourages deep explorative analysis of a narrowly defined set of phenomena, whereas wide-scope encourages a focus on the integration between the different linguistic levels, including their interfaces, similarities between the different levels, and their link to a unified linguistic theory. The lack of coordination between narrow-scope treebanks means that they may be based on mutually incompatible assumptions about the underlying linguistic structure, the division of labour between the different treebanks, and the choice of analyses for the phenomena where they overlap. This can make it difficult to

[2] The English CDT treebank includes a small 4,500 word CDT-annotated subset of the Penn Treebank, which can be used as a bridging treebank by treebank conversion systems.

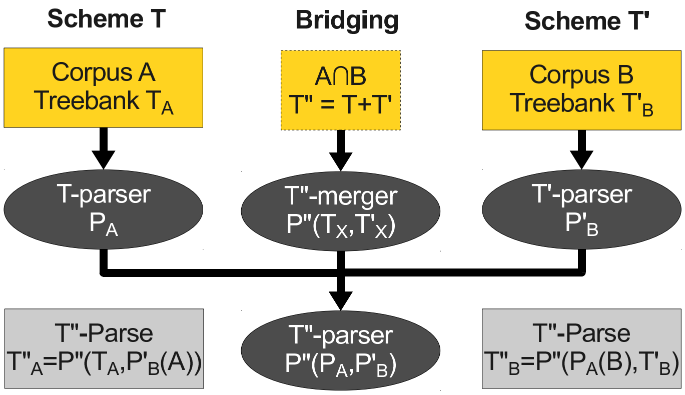

Figure 4: Annotation merging: Merging treebanks T_A, T'_B using parsers P_A, P'_B and merger P'' trained on gold-standard merge $T''_{A \cap B}$, resulting in T''-parser P''.

produce a coherent unified wide-scope treebank by merging several narrow-scope treebanks for different linguistic levels, as pointed out by Meyers et al [13, 11].

Since future applications in language technology are likely to require a coherent set of annotations at several linguistic levels, the integration between annotations at the different linguistic levels should be a key priority in future treebanking research — either as research in unified wide-scope annotation, or as research in systems for annotation merging. The GLARF system proposed by Meyers et al [11] is partly rule-based and designed for a specific set of treebanks. It would therefore be desirable with research in probabilistic general-purpose annotation merging systems that could be trained on two (or more) treebanks T_A, T_B on the basis of a small bridging treebank $T_{A \cap B}$, as shown schematically in Figure 4.

Challenge 3: Building a multi-parallel “bridging” corpus community

The researchers in the field should agree on a balanced, general-purpose, mixed-genre English corpus to be used as the English component in a collaborative multi-parallel “bridging” corpus. National research groups would be responsible for translating the English corpus into their own language and contributing their annotations. If these translations formed a substantial part of the corpora used in national treebank projects, it would be a lot easier to merge annotation systems and transfer annotations from one language to another.[3]

The English source corpus should be constructed with great care. It should be composed so that it is suitable for annotation at all linguistic levels (morphology, syntax, discourse, anaphora, semantics), with a permissive license (preferably an open-source license) that places as few restrictions on the subsequent use as pos-

[3] Translations are known to be coloured by the source language, so national treebank projects cannot be expected to work on translation corpora exclusively. But since translation to and from English is one of the major applications for language technology, the decision to include translations as a substantial part of the annotated corpus would make sense, also from a purely national perspective.

sible. To accommodate the needs of different treebank projects large and small, the English source corpus should be diverse, balanced, mixed-genre and general-purpose, and structured as an onion: there should be a tiny core corpus (say, 1,000 words) consisting of isolated sentences or small text excerpts chosen for their linguistic variation, supplemented with a wide range of larger corpora (say with 3k, 10k, 30k, 100k, etc. up to 10M words) that extend the smaller corpora by expanding the existing text excerpts and adding new text excerpts from more texts.[4]

It is important that the bridging corpus is accepted by the researchers in the field as the standard base corpus in most annotation projects (unless there is good reason otherwise), so the decision procedure must be thought out carefully. Eg, perhaps a community committee can specify a set of desiderata for the English source corpora. Different research groups can then come up with competing proposals, which the entire treebanking community can vote on — with a possible revote after a phase where the best proposals are merged.

Challenge 4: Linking treebanks with linguistic theories

When creating a treebank, the linguistic annotations are not observable quantities, but theoretical constructs informed by the annotators' conception of linguistic theory coupled with their intuitions about the language and the text. To be meaningful, annotations must therefore be interpreted within some notion of linguistic theory. We obviously need to be careful: we cannot corroborate a theory with annotations that presume the theory, which is why some annotation projects seek to be "theory-neutral", ie, to avoid basing the annotations too closely on linguistic theory. On the other hand, a theory-neutral approach is not bullet-proof, and it takes more than a few counter-examples in a theory-neutral annotation to disprove a theory: after all, the counter-examples might just be artifacts of ill-conceived annotation guidelines or misjudged intuitions by the annotators.

From a methodological point of view, I think the best solution to this dilemma is to give the annotations a clear interpretation in terms of linguistic theory, but allow annotators to mark cases where the theory is hard to apply. During the annotation, this will promote a rich interaction between the annotation and the linguistic theory. Moreover, without a clearly formulated interpretation of the annotations, it is difficult for other researchers to criticize the annotations or the underlying theory. The Copenhagen Dependency Treebanks may serve as an example of the approach that I am advocating. The annotations are heavily informed by the dependency theory Discontinuous Grammar [2], which stipulates how dependency structures determine the word order and the compositional semantics. The linguistic theory has been a huge help in the design of the annotation scheme. Since the theory is much cleaner if discourse structure is viewed as the continuation of sentential syntax to the inter-sentential level, a tree-structured discourse supplemented with coreference relations has been a guiding principle in our discourse annotation (a

[4] The "Pie in the Sky" corpus [12] may serve as an inspiration.

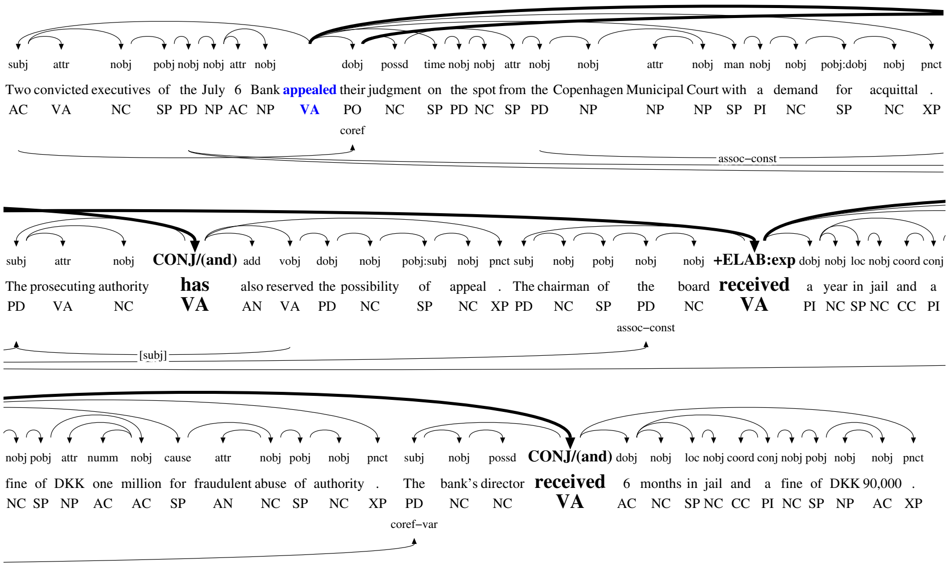
subj attr nobj pobj nobj nobj attr nobj dobj possd time nobj nobj attr nobj nobj attr nobj man nobj nobj pobj:dobj nobj pnct
Two convicted executives of the July 6 Bank appealed their judgment on the spot from the Copenhagen Municipal Court with a demand for acquittal .
AC VA NC SP PD NP AC NP VA PO NC SP PD NC SP PD NP NP NP SP PI NC SP NC XP
coref
assoc−const
subj attr nobj CONJ/(and) add vobj dobj nobj pobj:subj nobj pnct subj nobj pobj nobj nobj +ELAB:exp dobj nobj loc nobj coord conj
The prosecuting authority has also reserved the possibility of appeal . The chairman of the board received a year in jail and a
PD VA NC VA AN VA PD NC SP NC XP PD NC SP PD NC VA PI NC SP NC CC PI
assoc−const
[subj]
nobj pobj attr numm nobj cause attr nobj pobj nobj pnct subj nobj possd CONJ/(and) dobj nobj loc nobj coord conj nobj pobj nobj nobj pnct
fine of DKK one million for fraudulent abuse of authority . The bank's director received 6 months in jail and a fine of DKK 90,000 .
NC SP NP AC AC SP AN NC SP NC XP PD NC NC VA AC NC SP NC CC PI NC SP NP AC XP
coref−var

Figure 5: Example of a CDT syntax-discourse annotation. The primary dependencies form a tree structure shown on the top, the bottom arcs encode additional secondary dependencies and coreference links needed for semantic interpretation.

similar principle applies to morphology, with some adjustments that deal with non-concatenative morphology [5, 14]). The annotators have encountered hard cases, which have sharpened the theory by requiring revisions in theory and analysis. But the basic assumption about a tree-based discourse structure has held up, which has allowed us to contribute to the long-standing theoretical debate about whether discourse structure is best viewed as a tree or as a graph [4]. Figure 5 shows a unified CDT annotation of syntax and discourse.

The new data-driven statistical paradigm in computational linguistics has many virtues, but it has also led to an unhealthy decline in interactions with theoretical linguistics: treebanks and other linguistically annotated corpora are now the only place where linguistics really comes into play. In my view, one of the main challenges for the treebanking community is to build a stronger two-way interaction between linguistic annotation and linguistic theory: the theory should provide sense and direction for the annotations as well as their application in natural language processing. This is the best way of using linguistic theory as a guide to simpler and more useful annotation, the best way of moving linguistic theory forward, and the best way of bringing linguistics back into computational linguistics.

Challenge 5: Quantifying treebank quality

Any treebank project is faced with a wealth of choices where there are good linguistic arguments for more than one choice. Even close collaborators working within the same annotation project may disagree about the right analysis. The wealth of annotation schemes in treebanking is therefore unsurprising. Ideally, these choices should be made in a more principled way, and in some distant future, treebank quality will perhaps be evaluated by the perplexity that a linguistically based language model trained on the treebank assigns to an unseen corpus, and hard design decisions in the annotation scheme will perhaps be made partly or fully on the basis of such scores. But so far, there is no easy solution in sight. Measuring treebank quality is probably one of the hardest and most important outstanding problems in the field, and any research that can adress these problems even tentatively should be encouraged by the field.[5]

Unfortunately, in the absence of a better measure, inter-annotator agreement seems to have taken up an unhealthy role as the primary measure of annotation quality. Reviewers routinely request agreement figures when reviewing treebank papers, and people have suggested that low κ values (say, below 0.8 or 0.67) makes inter-coder reliability so low as to leaving the annotations useless (cf. the excellent review article by Artstein and Poesio [1]). There is no doubt that agreement and related measures have important uses: annotation projects should keep constant track of inter-annotator agreement and confusion tables for the individual relations, and prompt annotators and linguists to try to eliminate major sources of disagreement. However, apart from that, it is not clear that agreement has a constructive role to play: Reidsma and Carletta [17] have shown that agreement is a poor predictor of the performance of a system based on machine learning; more importantly, if used as a proxy for annotation quality by treebank designers and reviewers, an exaggerated focus on agreement may lead to distortions in the way treebanks are designed.

Agreement and confusion scores are highly beneficial when used to identify misunderstandings and formulate linguistically well-motivated clarifications of the annotation scheme. The distortions happen when treebank projects design their annotation schemes so that they optimize agreement, regardless of whether the agreement-boosting measures fail to be linguistically motivated. As an extreme case, we can construct a dependency treebank with 100% inter-annotator agreement by picking an annotation scheme where every word is analyzed as a dependent of the preceding word, using a single dependency label; a parser trained on this treebank would have a 100% labeled attachment score. Slightly more subtle agreement measures that correct for chance agreement, like κ and α, will assign a low agreement score to this annotation scheme; but even they can be tricked to yield a near-100% score, if we subdivide our single label into several subtypes which are easy for humans (and parsers) to disambiguate. For example, we can use the word class (or some other easily inferable quantity) as our relation label — which is what the Copenhagen Dependency Treebank, and many other treebanks, inadvertently do when they use labels such as 'pobj' for prepositional objects, 'nobj' for nominal objects, etc. For this reason, chance-corrected agreement provides a false sense of

[5] Nilsson, Nivre and Hall [15] have made an interesting experiment where they show that simple treebank transformations (changing the structure of coordinations or verb groups) can improve MaltParser performance, but they also show that these transformations are suboptimal for the MSTParser, ie, their method cannot really be used to make an unambiguous case for one analysis over the other.

security, and does not produce comparable scores even when comparing treebanks with the same underlying corpus and number of labels.

Another way of boosting agreement is to instruct annotators to always use a particular ad-hoc analysis for ambiguous constructions, eg, preferring VP attachments over NP attachments when in doubt. This move increases agreement, but also induces an arbitrary bias which may be harmful in some applications. Semi-automatic annotation also boosts agreement because annotators are biased towards accepting the analysis proposed by the parser, unless it is clearly wrong. Even worse, we may decide to simplify the annotation scheme by merging linguistically well-motivated labels that are often confused with each other, such as dependency relations that reflect fine-grained semantic distinctions (adverbial relations, discourse relations), and so on. The experience of the CDT annotators, and many others in the field, is that semantic distinctions are really hard to make, and that disagreements are often caused by truly ambiguous texts where the two differing analyses either lead to essentially the same meaning, or the context does not contain sufficient information about the speaker's true intentions. But that does not necessarily imply that the distinction does not encode important information, it is just noisy information.

The big question is: do we really improve treebank quality by making linguistically unmotivated design decisions that improve agreement, but reduce informativity? There are many desiderata for a good measure of treebank quality, but one of the more important is that it should be impossible to improve the quality score by merely throwing away information, for example, by merging labels mechanically. Agreement clearly fails on this criterion. Perhaps chance-corrected agreement measures can be fixed in part by measuring agreement using the highest-scoring set of merged labels, rather than the unmerged set of labels, but it is not clear that the resulting score would be interesting.

Until we get a better measure of treebank quality, reviewers should probably focus less on total agreement and more on a qualitative assessment of the confusion table, which encodes the probability $\mathrm{Conf}(l'|l)$ with which the other annotators used label l' when one of the annotators used label l. In the CDT project, 10% of the annotated texts are double-annotated: this allows us to compute a confusion table, which is included in the CDT annotation manual [3]. As an illustration, Figure 6 shows some of the confusion scores for the syntactic relations in the ongoing CDT annotation: "Agr" specifies the relation-specific agreement, ie, how often did the annotators agree on the label when one of them used the label; "N" specifies the number of tokens for which one of the annotators used the label; and "SN1" specifies the primary signal-to-noise ratio, defined as the ratio between the probability that the other annotators used the same label relative to the probability that the other annotators used the most frequent alternative label. In a classification task, the SN1 ratio can be expected to show a better correlation with machine learning success than agreement, since most classifiers will pit the two highest-ranked labels against each other, ie, the label will be hard to learn if the ratio is smaller than 1. Perhaps an even better predictor of classification success can be constructed by

Rel	Agr	SN1	N	Confusion list
expl	86%	8.6	53	expl$_{86\%}$ subj$_{10\%}$ preds$_{1\%}$ time$_{0\%}$ pobj$_{0\%}$
focal	42%	3.2	38	focal$_{42\%}$ attr$_{13\%}$ other$_{8\%}$ pnct$_{6\%}$ loc$_{5\%}$ nobj$_{3\%}$ err$_{2\%}$ correl$_{2\%}$ eval$_{2\%}$ mod$_{2\%}$ pobj$_{2\%}$ subj$_{1\%}$ dobj$_{1\%}$ ...
iobj	63%	2.4	19	iobj$_{63\%}$ dobj$_{26\%}$ robj$_{5\%}$ pnct$_{1\%}$ subj$_{1\%}$ nobj$_{0\%}$ attr$_{0\%}$ possd$_{0\%}$ modp$_{0\%}$
conc	21%	1.8	23	conc$_{21\%}$ contr$_{13\%}$ mod$_{13\%}$ prg$_{8\%}$ other$_{8\%}$ pobj$_{6\%}$ nobj$_{6\%}$ conj$_{5\%}$ attr$_{5\%}$ pnct$_{4\%}$ dobj$_{2\%}$ subj$_{2\%}$ possd$_{1\%}$ appr$_{0\%}$
iter	19%	0.4	26	time$_{46\%}$ iter$_{19\%}$ other$_{7\%}$ vobj$_{5\%}$ attr$_{3\%}$ eval$_{3\%}$ mod$_{3\%}$ nobj$_{1\%}$ dobj$_{1\%}$ relr$_{1\%}$ cause$_{1\%}$ name$_{1\%}$ ...

Figure 6: Some confusion scores from the CDT annotation manual.

taking the frequencies with which the different labels occur into account.

Although the CDT annotation is still ongoing and we hope to improve the inter-annotator consistency, it is worth noting that even labels with a high degree of confusion contain a lot of information, which will be lost if we start merging labels to improve agreement. By releasing the confusion table along with the treebank, the decision about which labels to merge can be left to the users of the treebank. However, when allowing a higher level of disagreement in the treebank, we also have to reconsider how we score parsers trained on the treebank. That is, the parser must get a score of 1 if it produces time or iter when the gold standard says iter, but a score of 0 if it produces subj. For example, a parser that produces label l' when the gold standard has label l can reasonably be scored with:

$$\max\left(1, \frac{\mathrm{Conf}(l'|l)}{\mathrm{Conf}(l|l)}\right)$$

It is quite possible that there are better ways of using the confusion table to score data-driven systems. The central point here is that the current focus on agreement in treebanking is unfortunate because it has unintended side effects in terms of what people decide to annotate and how they design their annotation schemes: increasing agreement by increasing bias with ad-hoc rules or losing informativity is not necessarily what we need most at the present state of our science.

3 Conclusions

With the advent of data-driven systems, linguistic annotation has become a great success and is maturing as a field. There is however still a number of important unsolved challenges. Most of them are concerned with how to exploit synergies by linking up different annotation projects with each other, even when they use different base corpora, focus on different linguistic levels and different languages, and are based on different conceptions of linguistic theory. Designated data-driven bridging tools coupled with collaborative bridging corpora are probably the key to long-term success in this area.

The wealth of different annotation schemes suggest that we need more research in how we assess the quality of linguistic annotations, and how we compare competing annotation schemes. At the same time, we must be careful to avoid that our measures of annotation quality do not lead to unintended incentives that distort what people annotate and how they design their annotation schemes. In particular, the current focus on inter-annotator agreement is probably unfortunate because the scores are hard to compare and encourage information loss and bias in the form of linguistically unmotivated ad-hoc principles. Finding better measures of annotation quality is therefore a key priority for the field.

4 Acknowledgments

My research was supported by a grant from the Danish Research Council for the Humanities. Thanks to my colleagues from the Copenhagen Dependency Treebank Project for many inspiring discussions, and to the workshop organizers for their valuable help.

References

[1] Ron Artstein and Massimo Poesio. Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(3), 2008.

[2] Matthias Buch-Kromann. *Discontinuous Grammar. A dependency-based model of human parsing and language learning*. VDM Verlag, 2009.

[3] Matthias Buch-Kromann, Morten Gylling-Jørgensen, Lotte Jelsbech Knudsen, Iørn Korzen, and Henrik Høeg Müller. The Copenhagen Dependency Treebank repository. http://code.google.com/p/copenhagen-dependency-treebank, 2010.

[4] Matthias Buch-Kromann, Iørn Korzen, and Daniel Hardt. Syntax-centered and semantics-centered views of discourse. Can they be reconciled? In *Proceedings of the DGfS 2011 workshop Beyond Semantics (to appear)*.

[5] Matthias Buch-Kromann, Iørn Korzen, and Henrik Høeg Müller. Uncovering the 'lost' structure of translations with parallel treebanks. In Fabio Alves, Susanne Göpferich, and Inger Mees, editors, *Methodology, Technology and Innovation in Translation Process Research*, volume 38 of *Special issue of Copenhagen Studies of Language*, pages 199–224. 2009.

[6] David Chiang. Learning to translate with source and target syntax. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 1443–1452, Morristown, NJ, USA, 2010. Association for Computational Linguistics.

[7] Silvie Cinková, Josef Toman, Jan Hajič, Kristýna Čermáková, Václav Klimeš, Lucie Mladová, Jana Šindlerová, Kristýna Tomšů, and Zdeněk Žabokrtský. Tectogrammatical annotation of the Wall Street Journal. *Prague Bulletin of Mathematical Linguistics*, 92, 2009.

[8] Richard Johansson and Pierre Nugues. Extended constituent-to-dependency conversion for English. In *Proceedings of NODALIDA 2007*, 2007.

[9] M. Marcus, G. Kim, M. Marcinkiewicz, R. MacIntyre, A. Bies, M. Ferguson, K. Katz, and B. Schasberger. The Penn Treebank: Annotating predicate argument structure. In *ARPA Human Language Technology Workshop*, 1994.

[10] André F. T. Martins, Dipanjan Das, Noah A. Smith, and Eric P. Xing. Stacking dependency parsers. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 157–166, Morristown, NJ, USA, 2008. Association for Computational Linguistics.

[11] A. Meyers, M. Kosaka, N. Xue, H. Ji, A. Sun, S. Liao, and W. Xu. Automatic recognition of logical relations for English, Chinese and Japanese in the GLARF framework. In *SEW-2009 at NAACL-HLT-2009*, 2009.

[12] Adam Meyers. Introduction to Frontiers in Corpus Annotaton II: Pie in the Sky. In *Proc. of ACL 2005 Workshop: Frontiers in Corpus Annotation II: Pie in the Sky*. 2006.

[13] Adam Meyers. Compatibility between corpus annotation efforts and its effect on computational linguistics. In Paul Baker, editor, *Contemporary Approaches to Corpus Linguistics*. Continuum Publishers, 2009.

[14] Henrik Høeg Müller. Annotation of morphology and NP structure in the Copenhagen Dependency Treebanks. In *The Ninth International Workshop on Treebanks and Linguistic Theories*, 2009.

[15] Jens Nilsson, Joakim Nivre, and Johan Hall. Generalizing tree transformations for inductive dependency parsing. In *Proc. ACL-2007*, 2007.

[16] Joakim Nivre and Ryan McDonald. Integrating graph-based and transition-based dependency parsers. In *Proceedings of ACL-08: HLT*, pages 950–958, Columbus, Ohio, June 2008. Association for Computational Linguistics.

[17] D. Reidsma and J. Carletta. Reliability measurement without limits. *Computational Linguistics*, 34(3):319–326, 2008.

[18] Anders Søgaard and Christian Rishøj. Semi-supervised dependency parsing using generalized tri-training. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 1065–1073, Beijing, China, August 2010.