**Çağri Çöltekin and Ben Campbell: TüBa-DZ to UD conversion and UD error detection**

This presentation covers two studies related to Universal Dependencies. The first part of this presentation will summarize the automatic conversion of the TüBa-DZ constituency treebank into Universal Dependencies. The conversion process will be discussed, including dependency extraction, POS conversion, head-finding rules, topological fields, as well as unusual cases that were unable to be properly converted into the Universal Dependencies annotation scheme.

The second part covers error detection for Universal Dependencies treebanks for three languages: English, German, and Finnish. The approach used was to compare like strings in the treebanks and see where discrepancies in the dependency relations occurred. In general there were three types of results: 1) false positives, ie. discrepancies that turned out not be errors when taking into account the larger context of the sentence, 2) true positives, ie. clear error(s) in the annotation of at least one of the sentences, and 3) borderline or ambiguous cases where two or more of the differing dependency relations could be argued to be correct and stricter guidelines would be necessary in order to make a clear determination of what the correct dependency relation is.

**Daniël de Kok and Tobias Pütz: Task-specific pre-training for dependency parsing**

Context-sensitive word representations (ELMo, BERT, XLNet) have provided large improvements over co-occurence based word embeddings accross various tasks. Context-senstitive word representations use deep models that are trained on an auxiliary objective, such as language modeling or masked word prediction. An important criticism of context-sensitive word representations is the excessive amount of computation time that running both training and inference with such models require.

In this talk, we discuss an alternative to such word representations in the form of task-specific pretraining. Task-specific pretraining uses the objective of a specific task and a corpus that was automatically annotated using a (weaker) model for that task. We show that in dependency parsing, task-specific pretraining results in models that achieve labeled attachment scores similar to models that use context-sensitive word representations. However, it achieves such performance with models that are smaller, faster, and require far less compute to run training and inference.

We will also attempt to address the question what task-specific pre-trained models actually learn and how that differs from models that do not use pretraining through model introspection in the form of probing tasks and gradient analysis.

**Patricia Fischer: Modeling compatibility in neural dependency parsing**

Composition models of distributional semantics construct representations for phrases from distributional word representations. Such models can be used to compose any two words or phrases into larger phrases. In natural language, however, the set of words which can combine is limited. Compatibility models specify which words and phrases can be composed into grammatically and semantically correct phrases.

In more traditional analyses of dependency distributions, lexical preferences have been used to model compatibility. Bilexical preferences were shown to improve parsing results. Recent approaches to neural dependency parsing implicitly encode information about co-occurrences through vector representations of the token input. In my talk, I will show that lexical preferences also add information to neural dependency parsers.

For dependency parsing, parsing as a sequence labeling task has recently established a new state of the art. I will extend the analysis of lexical preferences to dependency parsing as sequence labeling by inspecting the ranking of attachment candidates in a sequence labeling model.

**Sebastian Pütz: Insights into subword embeddings**

Word embeddings with subword representations are a staple choice in NLP, but little has been done to explore how these models store information. In this talk, I will show that fastText-style models often leave a large partition of their vector space untrained and discuss how an explicit ngram lookup-table can alleviate this issue. Moreover, I will provide insights into how downstream tasks are influenced by changing the subword lookup.