CSE/LIN 467/567: Extra assignment report

Ziang Li

Bohang Hua

**Abstract**

The purpose of this report is to investigate the mechanism behind the syntactic analysis by using the Stanford Parser and its conversion between the Penn treebank corpus and the dependency structure.

The syntactic analysis determines whether the composition of the input word sequence (generally a sentence) conforms to a given grammar and constructs a syntax tree to determine the structure of the sentence and the relationship between the syntactic components of each level. It will determine which words in a sentence have formed a phrase, which words are the subject or object of the verb.

**1**           **Introduction**

Syntactic analysis is one of the key techniques in natural language processing. Its basic task is to determine the syntactic structure of a sentence or the dependencies between words in a sentence.

It mainly includes two aspects. One is to determine the grammatical system of the language, that is, to give a formal definition of the grammatical structure of legal sentences in the language; on the other hand, the syntactic analysis technique, that is, to automatically derive the syntactic structure according to the given grammatical system. In addition, analyze the syntactic units contained in the sentence and its relationship is one of the most important in the syntactic analysis.

The language we used for the analysis is Chinese. In Chinese, there are lots of types of syntax. Even for native speaker, it is hard to determine every syntax correctly without any errors. That is, the syntactic analysis must be a tough trial especially in Chinese language.

Therefore, to ensure we get the correct parsing result as much as possible, we use Stanford parsing.

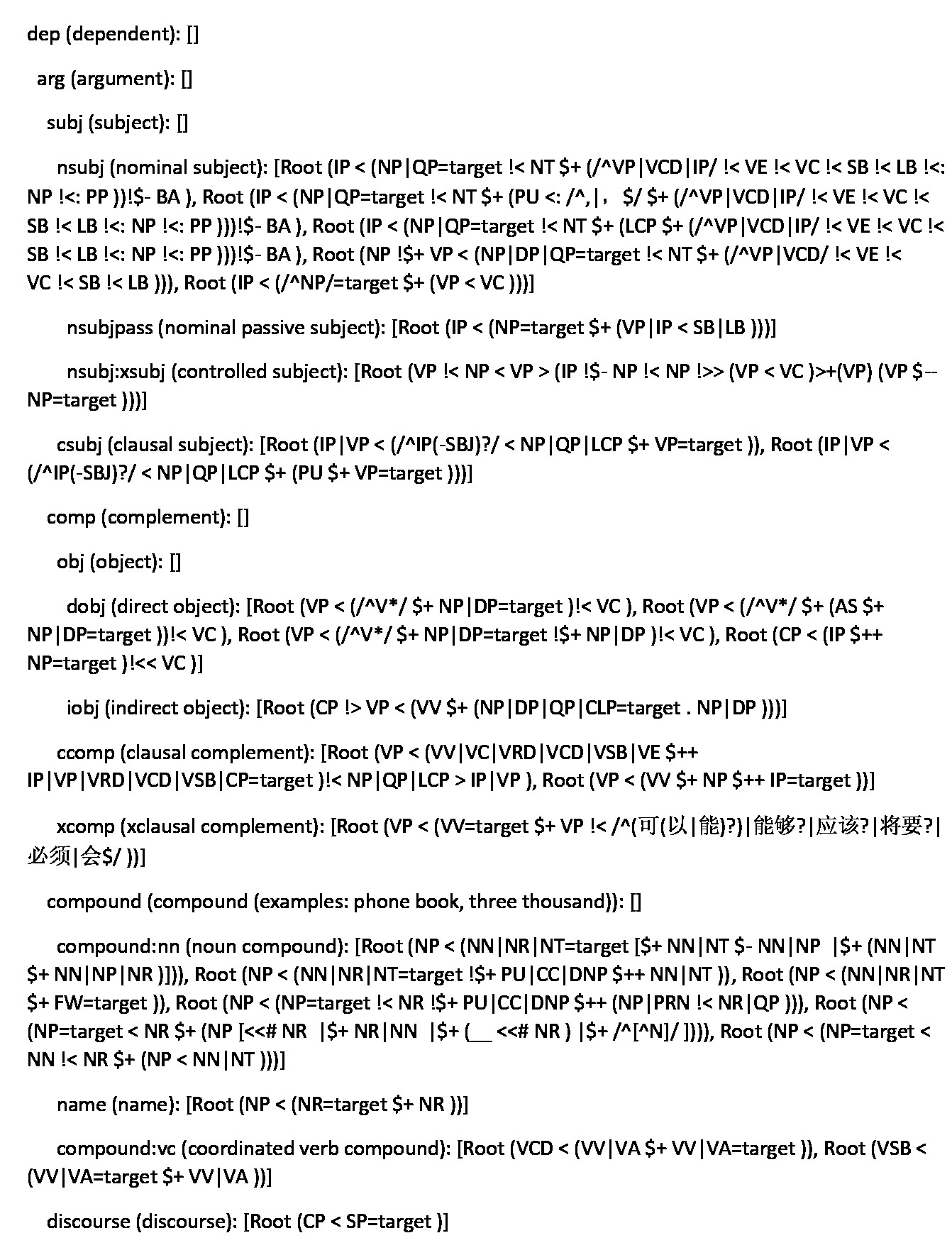
Stanford parser is an open source parser developed by the Stanford University Natural Language Processing Group and is a JAVA implementation based on probabilistic syntax analysis. The analyzer currently provides 5 Chinese grammars. Based on the authoritative and reliable Penn Treebank as the data of the analyzer, it has provided syntactic analysis functions.

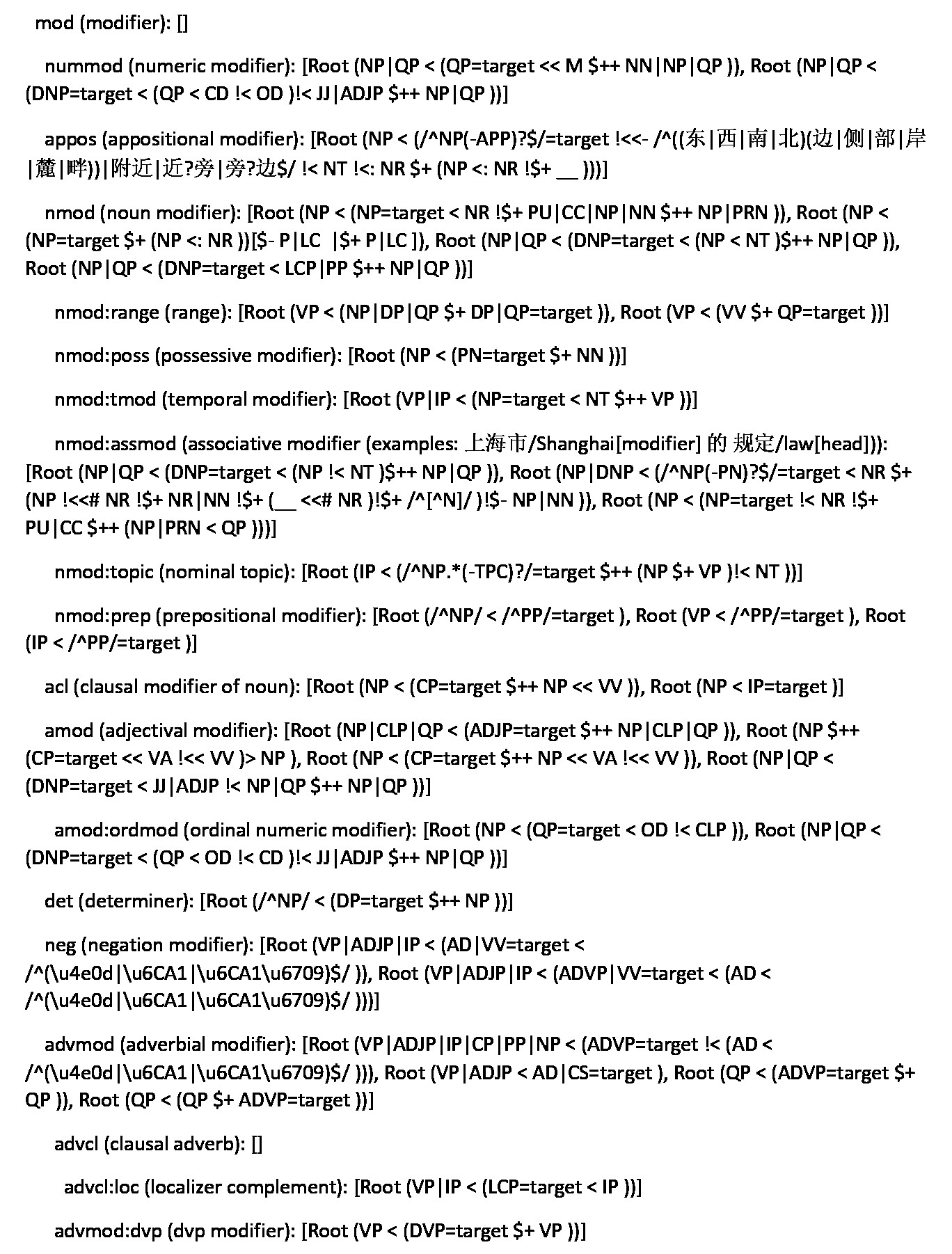
In addition, the analyzer has built-in word segmentation tools, part-of-speech tagging tools, and parser training tools based on custom tree libraries.

**2**         **Converting rules**

Here are the converting rules we use for the parsing Chinese. The actual code is listed in the following URL.

<https://drive.google.com/open?id=1hfo6ZvgVZeEX5WSuqcLTgSn7WbRhsOyK>







**3**         **Parsing results**

**3.1**         **Constituent parsing results**

During the constituent parsing, we found some problems in Chinese conversion.

1. The program is based on the English WSJ tree library. Causes loading tree to fail.

**Solution**: Change the name to chtb\_xxx.fid ==>wsj\_xxx.mrg. The renamed file is placed in a directory (such as NewCTBData).

1. Chinese has more SGML formats. Causes loading tree to fail.

**Solution**: Remove these tags.

1. Some sentence contains "(IP". Causes the split to fail.

**Solution**: Add "(" and ") before and after such sentences.

1. Some file contains (NP (NN 政府))))

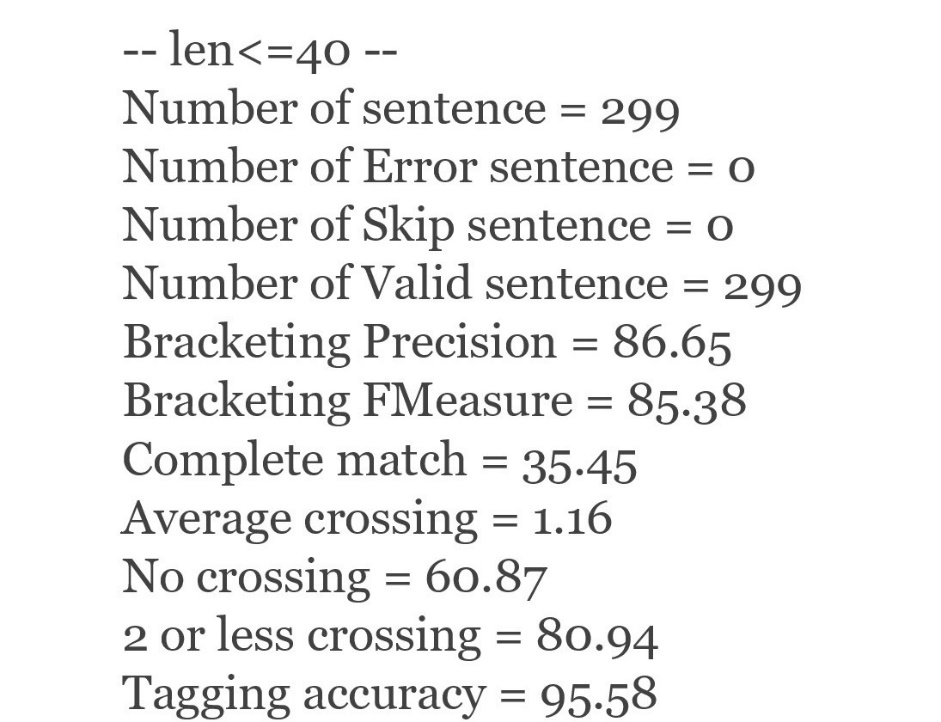
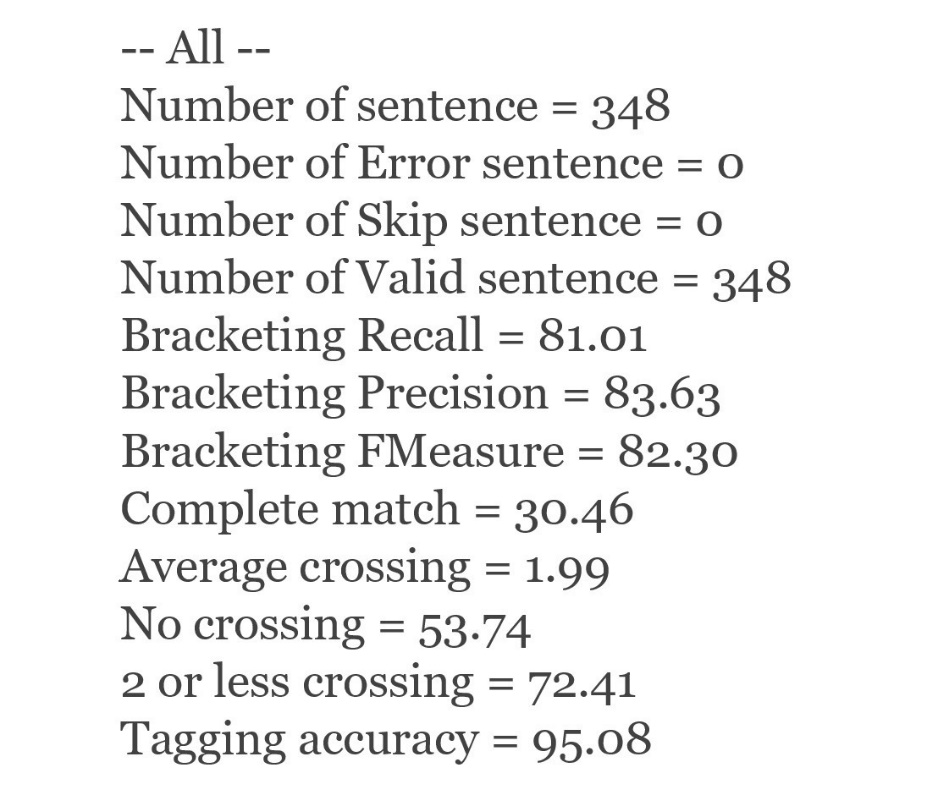
**Solution**. Revised to "(NP (NN 政府)))"

1. The program uses a fixed split according to the file ID. Train: 1-25, 26-270, 400-1151; Dev: 301-325; Test: 271-300

**Solution**: Replace the corresponding file, make sure that each segment must have at least one file. For example, 1-25 must have at least one file, and 26-270 must have at least one file.

With CTB5.1, the training set uses chtb001.fid-chtb270.fid + chtb400.fid-chtb1021.fid + chtb1030-chtb1151.fid (when the training set includes the file chtb1022.fid or chtb1129.fid, it cannot find many test sentences. The parse tree will get the (()) as the result).

Evaluation results under ctb5.1 in the case of SMcycles=5

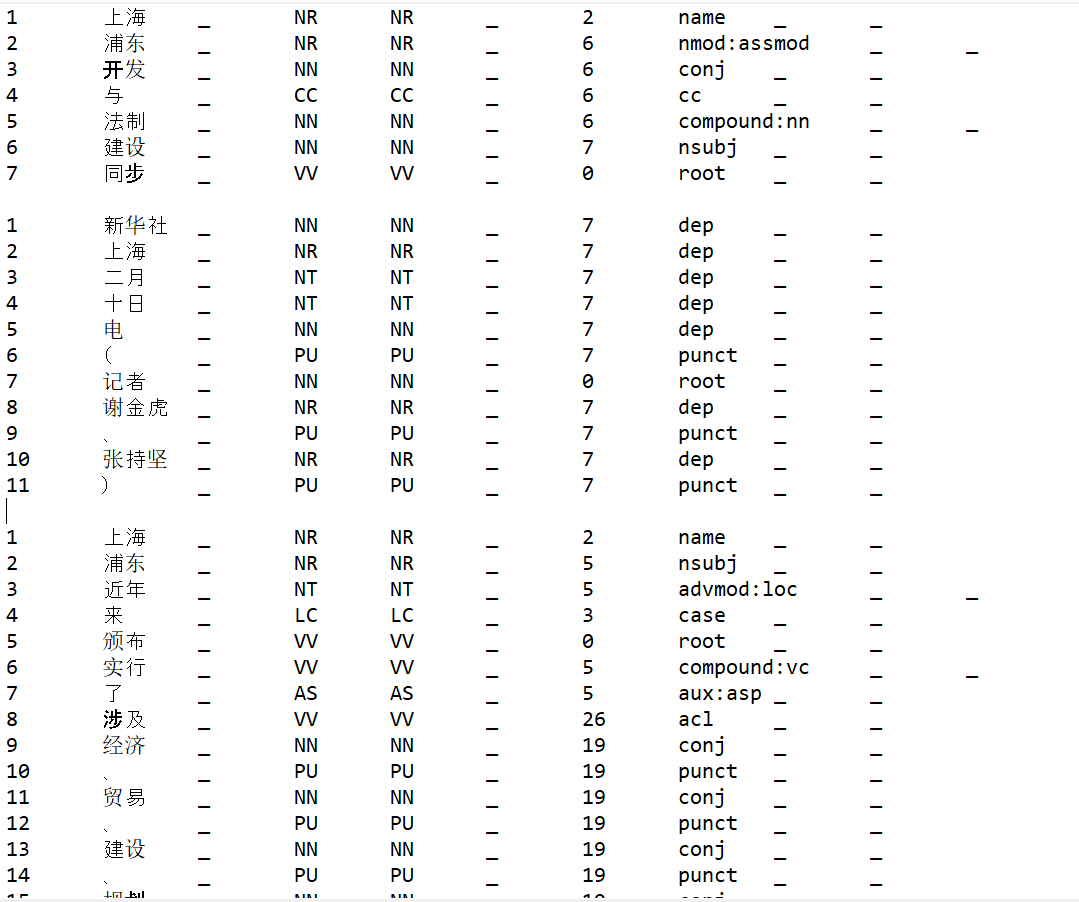


**3.2**        **Dependency parsing results**

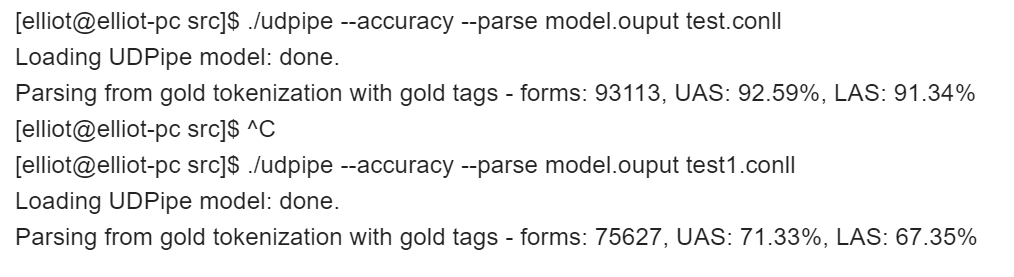
Here is the part of the result of dependency parsing. The complete result is listed on

the following URL.

<https://drive.google.com/open?id=1ajTbaDyiS6mfQgBNraWRKObQokmjpOyt>



We also use the udpipe to test the above result. Here is the result of udpipe testing.



**4**   **Discussion and conclusion**

**References**

de Marneffe, M.-C. and Manning, C. D. (2008). The Stanford Typed Dependencies Representation.

In *Coling 2008: Proceedings of the workshop on Cross-Framework and Cross-Domain Parser Evaluation*, pages 1–8, Manchester, UK. Coling 2008 Organizing Committee.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S., and McClosky, D. (2014). The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Balti- more, Maryland. Association for Computational Linguistics.

**A Data split**

**5 Difference between Stanford Dependency and Universal Dependency**