Middle East Technical University | Department of Computer Engineering

CENG 463

Introduction to Natural Language Processing

Fall 2020-2021

Assignment 3

Due date: January 30 2020, Saturday, 23:55

1 Objectives

In this assignment, you will train a dependency parser with the help of a library and compare your parser with existing parser models. You will additionally present a report that explains the concepts related to dependency parsing, present details of your training process.

2 Data

The data is available from Universal Dependencies organization. You will train your models on Turkish. The treebanks have a common annotation format. For more about the format of the data please refer to `http://universaldependencies.org/format.html`.

3 Specifications

3.1 Implementation

1. You will use the spaCy library for Python to train your models.
2. The data you are going to work with is provided by the Universal Dependencies treebanks for Turkish (UD_Turkish-MIST).
3. Take a look at the examples from the spaCy documentation and convert the data into an appropriate format. You are also strongly advised to check the guides provided by the spaCy.
4. Then train a dependency model with the train and dev segments of the data.
5. Run your models on the test segment and provide outputs in the CoNLLU format.
6. Use the official evaluation script in the CoNLL 2018 Shared Task's webpage to measure the success of your models.

3.2 Report

1. Explain projective and non-projective dependency parsing, give example sentences.
2. Compare projective and non-projective dependency grammars with context-free grammars.
3. Compare English and Turkish in terms of projectivity with example sentences. Also provide a comparison of the UD treebanks (UD English-ParTUT and UD Turkish-IMST) in terms of percentages and counts for projective and non-projective trees.
4. Explain transition-based and graph-based dependency parsing.
5. Explain labelled and unlabelled attachment scores.
6. Report where you would land on the scoreboard of CoNLL 2018 Shared Task.
7. Provide plots for iterations versus attachment scores for your training procedure.
8. Add a section reflecting on your implementation: what did you do step by step?

4 Regulations

1. **Programming Language:** You will use `Python3`.
2. **Late Submission** is not allowed.
3. **Cheating:** We have a zero tolerance policy for cheating. In case of a cheating event, all parts involved (source(s) and receiver(s)) get zero. People involved in cheating will be punished according to the university regulations. Remember that students of this course are bounded to the code of honour.
4. **Newsgroup:** `odtuclass`

5 Submission

- Submission will be done via `odtuclass`.
  Submit a `tar.gz` file named `assgn3.tar.gz` that contains all your files related to the assignment.