Second Year Project:
Natural Language Processing and Deep Learning
Week 5

Barbara Plank, Rob van der Goot

February 25, 2020

This is your assignment for week 5 of the course *Second Year Project*. You are expected to work on Part I - Task A on Tuesday and finish Part I - Task B on Friday. Similarly, you can start with Part II on Tuesday and finish it by the end of the week. Good luck!

After completing the whole assignment, you should:

- be familiar with the Universal POS tagset and data annotation for POS tagging
- be able to discuss annotation quality, both qualitatively and quantitatively, by comparing your annotations to those of a peer
- be able to implement a continuous bag of words embeddings (CBOW) encoder for lyrics era classification
- be able to improve your system with a model of your choice, and submit the predictions on held-out test data in a specific format

Requested reading: Chapter 8 of *Speech and Language Processing* by Jurafsky and Martin [1] and Chapter 6 of Pustejovsky and Stubbs [2].

Important dates

- Task 1: Submit the annotations of the tweets by **Wednesday Feb 26, 23:59 on LearnIT** (Part I Task A). This way you will get the annotations of your peer in the Friday lab.
- Task 2: Submit the predictions of your best system by **Sunday March 1, 23:59 on LearnIT** (Part II). Make sure your prediction file is in the expected format. This way we will provide you with the results on Tuesday.

1 Part I: Annotation and Annotation Quality

In this assignment we will (TASK A) annotate a small amount of data, and study inter-annotator agreements (task B). For this we will use the UPOS tag-set as discussed in the lecture.

Your task is to:

- Task A: annotate 20 tweets.
- Task B: evaluate the quality of this annotation and inspect common confusions. Evaluate the quality both qualitatively (discuss differences) and quantitatively (calculate Cohen's κ). This can only be done on Friday, as you need annotations from other annotators.

A: Annotation

Find the file with your student number in `assignments/week5/pos-data/`. In this file, you will find 20 tweets which are pre-tokenized. Behind each word you are supposed to annotate the pos tag, with one whitespace in between. The final file should look like this:

```
# text = new pix comming tomoroe
new ADJ
pix NOUN
comming VERB
tomoroe NOUN
```

You can use a whitespace or a tab between the word and its tag. Please check with the script `posCheck.py` whether the file format is correct. Usage: `python3 posCheck.py origFile annotatedFile`

For annotation guidelines we refer to the slides and https://universaldependencies.org/u/pos/all.html. Alternatively, it might be helpful to look at example annotations, which are provided in: `assignments/week5/pos-data/ewt-examples.pos.txt`.

Some domain specific decisions that are not covered in the guidelines:

- Usernames are PROPN
- Hashtags are X, unless used syntactically (e.g. “I am #working”)
- smileys are SYM
- ‘RT’ is X, unless used syntactically (e.g. “let me RT him”)

Upload your annotated file in the assignment on LearnIT. Please name your file like: `12345.pos.ann.txt` (replace *12345* with your student number). **If you don’t know your student number, ask the TAs in a direct message on Slack**.

B: Annotator Agreement

(this part can only be completed on Friday, as you need annotations from other annotators)

- Calculate the accuracy between you and the other annotator, how often did you agree?
- Now implement Cohen’s Kappa score, and calculate the Kappa for your annotation sample. In which range does you Kappa score fall?
- Take a closer look at the cases where you disagreed with the other annotator; are these disagreements due to ambiguity, or are there mistakes in the annotation?

2 Part II: Shared Task - Lyrics Era Prediction

In the assignment of week 3, you built a Naive Bayes classifier to classify heavy metal songs into a time-era based on their lyrics. In this week’s assignment we will make use of the same data. However, this time a secret test set is included, called `lyrics-data/metal-lyrics-test-hidden.csv`

2.1 CBOW

Implement a text classifier which uses the continuous bag-of-words encoding (discussed in lecture 7) for lyrics era prediction (sum over embeddings of the words). You are welcome to test any variation of the CBOW model. Compare your CBOW model performance to the NB model of week 3.

2.2 Shared Task

There are many ways in which you can improve the previously built classifiers. Many of these have been discussed during the course:

- Combine/merge the output of your classifier with other types of classifiers
- Optimize pre-processing

- Parameter-tuning
- Use of external data (word embeddings, POS tagger, ...)
- You are allowed to add the development data to the train data for the final run

Note: Scraping more gold data and training your model on that is *not* allowed.

Try to improve your classifier to obtain a better performance. Finally, run your model on the secret test data. Submit your file, where you replace `hidden` in the first column with your model's prediction (make sure to retain the order of the data).

Make sure that your system outputs exactly the right format. To facilitate this, we prepared a script to check your prediction file before submitting it. You can use it like:

```
python3 submissionCheck.py metal-lyrics-test-hidden.csv mysubmission.csv
```

Upload your annotated file in the assignment on LearnIT. Please name your file like: `test-12345.csv` (replace 12345 with your studentnumber).

References

[1] Jurfasky and Martin, In Preparation. *Speech and Language Processing (3rd ed. draft)*. Available at https://web.stanford.edu/~jurafsky/slp3/

[2] Pustejovsky and Stubbs. *Natural Language Annotation for Machine Learning*. O'Reilly, 2013.