Ethics for NLP: Spring 2022
Homework 3

TECHNISCHE UNIVERSITÄT DARMSTADT

Due until Thursday, 23.06. at 11:30am

Submission Guidelines for Homework

- This homework worth 20 Points.
- Use the provided .ipynb template to write your code and to answer the questions.
- Name your submission file as: hw03.<matriculation_number>.lastname.firstname.ipynb
- No need to submit data files, since we already have it.
- Extra credit shall be given to well-structured submissions.
- In case of questions or remarks, please contact:
  - Aniket Pramanick, pramanick@ukp.informatik.tu-darmstadt.de

1 Low Resource Languages (20 Points)

Before you start make sure you read the README.md file associated to this assignment for important setup information. Note that you don't need to use any additional libraries for this homework since they are already imported.

1.1 Goals

Not all languages in the world are widely used and supported for the natural language processing. In this homework you will be dealing with the part of speech tagging for languages where the available labeled data is limited.

1.2 Models for sequence tagging

The main task of sequence tagging is the prediction of the sequence $\mathbf{y} = (y_1, \ldots, y_n)$ for any given sequence of tokens $\mathbf{x} = (x_1, \ldots, x_n)$, where $y_i \in (1, \ldots, L)$, the labels of our interest. We will be using a scoring function s for the modelling of any sequence tags for the input sequence. With the scoring function the best prediction of the model can be realized as:

$$\hat{y} = arg\,max_y s(y, x) = arg\,max_y \sum_{i=1}^{n} \psi(y_i, i, x)$$

so that the classifier can use any of the features of the input sequence x and the position i when predicting the label y_i. To generate the features for the input sequence h_i you can find a bidirectional LSTM model in the provided .ipynb template. On top of this model we will apply a feed-forward layer to predict the POS tag at every step i.

1.3 Data

You will evaluate your models using the treebank of Universal Dependencies[1] - a framework for consistent annnotation of grammar accross different human languages. For this exercise, only part of speech tagging is

[1] https://universaldependencies.org/

relevant. You will also evaluate your model on 8 languages. Some are low resource, some are high resource languages. We define languages with more than 60K tokens in their dataset as high-resource and others as low-resource. The languages from the following language families will be considered:

- Germanic: English (en), Afrikaans (af)
- Slavic: Czech (cs)
- Romance: Spanish (es)
- Semitic: Arabic (ar)
- Baltic: Lithuanian (lt)
- Armenian: Armenian (hy)
- Dravidian: Tamil (ta)

1.4 Model

We will provide you a implementation of a BiLSTM model with the necessary data written in Pytorch. In the *saved_models* folder you can also find a model trained on English data.
Some helping comments, that may help you to run the model and to understand the code, are placed in the corresponding *.ipynb* template for this homework.

1.4.1 Tasks

After reviewing the provided code you should be able to run the model in the *train* and *evaluation* mode.

1.4.1.1 Task I: Training and evaluation (10 Points)

For this task, make sure you understand, which parameters does the function *run* accepts and how you can run it in different modes. First, evaluate the provided English model. Then, train the model for the following languages and evaluate it:

- Czech (cs)
- Spanish (es)
- Arabic (ar)
- Afrikaans (af)
- Lithuanian (lt)
- Armenian (hy)
- Tamil (ta)

1.4.1.2 Task II: Discussion (10 Points)

Now you have all the necessary inputs to discuss the differences between low and high resource languages. Please answer the following questions:

- How the performance changes across language families and available dataset size? Make a conclusion of how the model's prediction depends on the available data.

- What role does the training set size plays for the model? Which problem regarding training sets occurs when you deal with the low-resource languages?
- What do the parameters `n_layers`, `bidirectional` and `dropout` (variable `params` in the first code cell) of the LSTM model mean? According to your research results please answer the following questions regarding the low-resource languages:
  - What happens when you increase the variable `n_layers` and why?
  - What changes when the model is unidirectional and why?
  - What happens when you increase the dropout and why?
- Define the term “label noise”. After that, answer what happens if the label in the training data are noisy?