July 11, 2023

Natural Language Processing

Project 2.5: English-Czech Machine Translation

Organization

- Please read the project description carefully. All the details are presented in this document.
- Submit your solution no later than July 31st, 23:55. Submit your solution using the corresponding ISIS section, where you have downloaded this document.
- You can make unlimited re-submissions until the deadline, but the latest received submission will be graded only.
- Submit your solution as a zip file with the following name format: 'NLP_project2_5_[Your Names separated by underscores].zip' containing your codes and report files.
- For those who decided to work on the project in a group, one submission of the solution would be enough.
- For each task, you see a percentage that shows the task's score in the whole project.
- Please use Python 3 for coding.
    - You are free to use any package/module/library for different tasks (except when it's clearly mentioned in a task to don't use available packages to solve the task).
- Please put comments on your code, wherever you think it would help to improve the readability and understandability of your code.
    - You can submit your codes as a .py file or Jupyter notebook
- You should submit a short report, describing your approach to solving different tasks and also provide the obtained results (e.g., the evaluation result of your models).
    - For the report please use the ACM proceedings template from here (https://www.acm.org/publications/proceedings-template). The Office Word and latex templates are provided on the page.
    - The report should be between 4 to 6 pages.
- There is a bonus task at the end of the document, as the name implies it is not a mandatory task, but you can take your time and solve it to get more scores.

Plagiarism Statement

Your project, including the report and the code, will be checked against other submissions in the class. We trust you all to submit your own work only. Copying someone else's code and report and submitting it with minor changes, will be treated as plagiarism.

If you have any further questions, please write to: salar.mohtaj@tu-berlin.de

Introduction:

Machine Translation (MT) or automated translation is a process in which computer software translates text from one language to another without human involvement. Today, machine translation works best in scenarios where a text needs to be conveyed in an understandable form in another language. Using deep neural networks, the performance, and accuracy of machine translation models have increased significantly in recent years.

In this project, you are going to develop your machine translation model to translate English text into Czech and vice versa.

Data:

The data for this project is based on “European Parliament Proceedings Parallel Corpus 1996-2011”. The Europarl parallel corpus is extracted from the proceedings of the European Parliament. It includes versions in 21 European languages: Romanic (French, Italian, Spanish, Portuguese, Romanian), Germanic (English, Dutch, German, Danish, Swedish), Slavik (Bulgarian, Czech, Polish, Slovak, Slovene), Finni-Ugric (Finnish, Hungarian, Estonian), Baltic (Latvian, Lithuanian), and Greek.

As mentioned before, in this project you will focus on English-Czech texts.

Please download the parallel data containing English and Czech sentences from here (or use this address if the hyperlink doesn’t work: https://www.statmt.org/europarl/v7/cs-en.tgz). You can find more details about the data at https://www.statmt.org/europarl/.

Tasks:

Task 1: Data exploration (10%)

For this task, you should extract some insights (i.e., some statistics and graphs) from the provided data. It could be the length differences between the two languages and also the number of sentences in the whole corpus. Don’t limit yourself to these examples and try to find more insights in the data. Please highlight some of the most important findings in your report. Please highlight some of the most important findings in your report.

Moreover, since the dataset is too large, randomly select the amount of data that fits your machine to train your models in the following steps (e.g., 10% of data) (data sampling).

Hint: if you faced codec errors in reading files, the “codecs” library could be used.

Task 2: Pre-processing (15%)

In this task, first, apply all the necessary pre-processing steps that you think would help to better prepare your data for the next steps. You don’t have to apply all the pre-processing tasks which are covered in the course. Regarding the report, you should briefly mention in your report why you decided to apply the chosen pre-processing steps (and why not the others).

Among all the pre-processing tasks, the following are recommended:

- lowercase the text
- strip empty lines and their correspondences
- remove lines with XML-Tags (starting with "<")

Task 3: Neural Machine Translation (45%)

In this task, you should do the following sub-tasks. Choose two evaluation metrics and report your results using these two metrics.

- Split data into train, validation, and test sets. Use 20% of the data as the test set.
- Develop an RNN-based sequence-to-sequence model (encoder-decoder) to translate English input into Czech text.
    - In your report describe your reasons for choosing the architecture that you are using for the task
    - Test the impact of different embedding models (e.g., Glove, Word2Vec, and …) on your model's performance.
    - Interpret the results of your model in the report. Does the length of text impact the performance of the model? What characteristic of sentences led to a better translation by the model?
- Change your input and target languages (Translate from Czech to English) and train your mode again
    - Compare the results in this step with the results from the previous step where English was the input language.
- Develop a character-based model that trains to translate characters into the target language. Compare the results of this model with the achieved results from the word-based models before.

Please report all the achieved results with the models in your report document. Moreover, describe the hyper-parameters of your neural network model in the report.

Task 4: Neural Machine Translation with Attention (30%)

In this task, the idea is to improve the performance of your models from the last step by using the attention mechanism in your model. Compare the achieved results with and without the attention mechanism in your report. Also, visualize the attention weights for a sample instance and highlight it in your report.

Bonus Task: Pivot Translation (+30%)

A pivot language, sometimes also called a bridge language, is an artificial or natural language used as an intermediary language for translation between many different languages – to translate between any pair of languages A and B, one translates A to the pivot language P, then from P to B.

For this task, you should develop a neural machine translation model for translating Czech to German, using English as the pivot language. You can download the German-English parallel corpus from here.