INFO2049 Assignment 2
Practical Project

A list of projects, grouped based on topics, is available below. Choose any one of them. Besides the basic implementation, you have to perform the additional experiments.

For computing resources, please use Google COLAB or GPU severs (e.g. those available via the CECI).

Organizational Details

Project to be done in groups of at most 3. As far as possible, keep the same group as Assignment 1 (Scientific Article Review).

Choose any of the projects in which you are most interested. Different groups can work on the same project (no 1st come 1st serve rule as was the case in Assignment 1).

Inform Ms. Jamar (Julie.jamar@uliege.be) via email (cc: ashwin.ittoo@uliege.be) of your selection. In your email, please also mention who your team members are, even if they are the same as for Assignment 1.

Submission Details

Submission via lolo@, under Assignment 2. It should include

- Source codes
- Small readme file describing specific instructions to run your system and the URL from where to download the dataset you used.
- Performance scores for various experiments (described later in this document)

Deadline: 12th Nov. 2020, 23:59

Project Topics

Machine Translation

Project MT1

- Implement an MT system using RNN.
- Your MT system can be trained on Wikipedia (dumps can be downloaded for free).
- You can work on any language pairs, but either the source or target should be English.

Experiments:

- Investigate the performance using various word embeddings: Word2Vec, GloVe, FastText.
- Investigate the performance using BERT as contextual embeddings
- Investigate the performance by including the attention mechanism of Bahdanau et al. https://arxiv.org/abs/1409.0473
- (Report & interpret all scores)

Project MT2

Implement an unsupervised word-level MT system, similar to the one discussed in the lecture. See Conneau et al. for details (https://arxiv.org/abs/1710.04087)

Experiments:

- Investigate the performance using various word embeddings: Word2Vec, GloVe, FastText.
- (Report & interpret all scores)

Text Summarization

Implement a seq2seq summarization system. As inspiration, see Nallapati et al. (https://openreview.net/pdf?id=gZ9OMgQWoIAPowrRUAN6).

As corpus, use the Gigaword corpus, as describe in Nallapati et al.

Experiments:

- Investigate the performance using LSTM vs. GRU
- Investigate the performance using various word embeddings: Word2Vec, GloVe, FastText.
- Investigate the performance using document level embeddings. See Wu et al. (https://www.aclweb.org/anthology/D18-1482/)
- (Report & interpret all scores)

Sentiment Analysis

Implement an RNN with attention for sentiment analysis. As inspiration, refer to the study of Letarte et al. (https://www.aclweb.org/anthology/W18-5429/) or that of Ambartsoumian & Popowich (https://www.aclweb.org/anthology/W18-6219/).

Experiments:

- Investigate the performance using LSTM vs. GRU
- Investigate the performance using various word embeddings: Word2Vec, GloVe, FastText.
- Investigate the performance using document level embeddings. See Wu et al. (https://www.aclweb.org/anthology/D18-1482/)
- Investigate the performance without attention
- (Report & interpret all scores)

Language Models

Implement an RNN-based language modeling system. Your system your model the languages in newspaper texts (either English or French), which can be crawled. Alternatively, you can model the language from the English Wikipedia dumps.

Experiments:

- Investigate the performance using LSTM vs. GRU
- Investigate the performance using various word embeddings: Word2Vec, GloVe, FastText.
- If possible (you might end up having an extremely large number of parameters), include attention in your model. As inspiration, see Salton et al. (https://www.aclweb.org/anthology/I17-1045/). Investigate
- (Report & interpret all scores)