Biomedical Data Science & AI

Exercise sheet 12

SoSe2025

Prof. Dr. Holger Fröhlich, Jonas Radermacher, Jiajun Qiu, Sebastian Schwick

Due date: 8. July

Exercise 1: RNNs, and Transformers Foundation (10 Points)

1. **RNN Basics: (2 Points)**
   - Explain the structure and functioning of a basic Recurrent Neural Network (RNN) cell. What are the key components that enable RNNs to process sequential data effectively?
2. **Comparative Analysis: LSTM vs. GRU vs. Classic RNN: (2 Points)**
   - Compare and contrast the architectures of Classic RNNs, GRUs (Gated Recurrent Units), and LSTMs (Long Short-Term Memory) in terms of their:
     i. Cell structure and flow of information.
     ii. Ability to handle the vanishing gradient problem.
     iii. Capacity to capture long-term dependencies.
     iv. Computational efficiency during training.
3. **Attention Mechanism in Transformers: (2 Points)**
   - Explain the attention mechanism used in Transformers:
     i. Define self-attention and its benefits compared to traditional attention mechanisms.
     ii. Describe the components: Query, Key, and Value vectors.
     iii. How are attention weights computed using softmax and masking?
     iv. Discuss multi-head attention and its role in capturing diverse dependencies.
4. **Transformer vs. RNNs (Classic RNN, GRU, LSTM): (2 Points)**
   - Discuss how the Transformer architecture addresses limitations of RNNs (Classic RNN, GRU, LSTM) in handling long-range dependencies and enabling parallelization during training.
5. **Application Scenarios: (2 Points)**
   - Provide examples of natural language processing tasks where each architecture (Classic RNN, GRU, LSTM, Transformer) is most suitable. Justify your choices based on their strengths and weaknesses.

Exercise 2: Sentiment Analysis of Drug Reviews Using BERT (15 Points)

In this exercise, we want to implement and fine-tune a pretrained BERT model. We will use this model for classifying patient drug experiences into negative (≤4), neutral (4<rating<7), or positive (≥7) sentiment using review text and rating-derived labels. **(Use random_state=42 and 10% of training and test data.)**

1. Load the drug review datasets, label the sentiment classes, and encode the labels. (Hint: use a label encoder) **(3 Points)**

2. Plot the class distribution. **(1 Point)**

3. Initialize the BERT model (Hint: use "boltuix/NeuroBERT-Tiny" and the AutoTokenizer). **(1 Point)**

4. Define a tokenizer function (use padding="max_length", truncation=True, max_length=512). **(1 Point)**

5. Write a classifier class for sentiment classification using the BERT model, a dropout layer and a fully connected layer. **(3 Points)**

6. Train and evaluate your model using the pytorch AdamW optimizer and a CrossEntropyLoss. Why do we use CrossEntropyLoss here? Record loss and accuracy for both the training and the validation set. **(3 Points)**

7. Use your finetuned model to predict the test set. Create a classification report (hint: use sklearn classification report). Discuss your results. **(3 Points)**