Language as Data.

Projects (WS 2024/25)

1. Project 2: Modelling Language

For this project, you will train two language models; one for each of the languages that you analyzed in project 1. Split your corpus into 80% training data, 10% development data, and 10% test data. Use the test data only for final evaluations.

Submit the project report as a pdf and the code in a zip archive through Stud.IP. The filenames should read *02_firstname-lastname_firstname-lastname_firstname-lastname.{pdf,zip}* with the first and last names of each group member. The reports are due on Monday evening (20:00 CET), **23rd of December**. We expect you to submit descriptions for all deliverables. If you run into any problems working on this project, describe the problem and/or give a partial analysis. If you need additional computing resources, contact us early. We can give you access to the cluster, but this won't work on short notice.

1.1. Deliverable 1: Pre-Processing

Describe the pre-processing steps applied to the data. Your description should maximize reproducibility, i.e., mention all necessary details including versions of the libraries that you used.

1.2. Deliverable 2: N-Gram Baseline

Train an n-gram model for each language as the baseline. Use an n-gram size of 3; if your computing resources allow it, you could also test larger n-gram sizes. Report perplexity and discuss whether the *Unknown Word Problem* poses a problem for the baseline.

1.3. Deliverable 3: Simple Neural Language Model

Train a simple neural language model. Monitor perplexity on the development data and perform early stopping to avoid overfitting. Discuss the learning curves and evaluate perplexity on the test data. Compare the results to the n-gram baseline and reflect on the differences between the two languages.

1.4. Deliverable 4: Analyze Model Variants

Modify your model and evaluate the effect of the modification on the results. Perform at least one modification for three of the following categories:

- Input representation: modify data pre-processing or embedding representation.

- Model architecture: modify hyperparameters or model structure.
- Training process: modify how the model learns from the data.
- Sampling: modify how the model generates output.

Describe the modification in detail and explain how it corresponds to the respective category. Present your results in meaningful tables and plots. Discuss the results and reflect on the differences between the two languages.

1.5. Deliverable 5: Evaluate Model Output

Design a setup for human evaluation for language A. Decide on at least two evaluation categories (syntactic fluency, semantic coherence, naturalness, contextual adequacy,...) and write down annotation guidelines, including the value ranges and examples.

Use at least 15 prompts and generate outputs from two different model configurations. Annotate the model outputs individually. Discuss the results and evaluate inter-annotator agreement.

If the language is not known to all group members, approximate annotation using additional resources such as lexicons or machine translation engines. Document the process properly and discuss how the varying language proficiency of the annotators affects the results. Provide the model outputs and the raw annotations in the appendix.

1.6. Deliverable 6: Project Summary

Provide a brief summary of your project's key findings and reflect on the work process.