AHLT Lab Project

1 Context

This laboratory project is an evaluable component of the course “Advanced Human Language Technologies (AHLT)” in the Master in Artificial Intelligence. Its main goal is to help you gain practical experience with realistic Natural Language Processing (NLP) problems, tools, and evaluation methodologies.

You will be addressing a real biomedical Information Extraction task that has been studied by the research community: the extraction of drug names and drug–drug interactions from scientific texts. The project is grounded on a real shared task from the SemEval-2013 evaluation campaign, specifically *Task 9: Extraction of Drug-Drug Interactions from Biomedical Texts*, which provides annotated data, evaluation scripts, and well-defined benchmarks.

You will be provided with working solutions and reference implementations. Your role is to understand how these systems work, experiment with them, and improve their performance in a controlled and well-documented way.

2 Task Description

The project focuses on two complementary NLP tasks that are central to many Information Extraction pipelines:

- Named Entity Recognition and Classification (NERC): identify spans of text that correspond to entity names (drugs in this case) and classify them into predefined categories. This task is formulated as a sequence labeling problem, typically using a BIO-style tagging scheme.
- Drug–Drug Interaction (DDI) extraction: determine whether an interaction between a pair of drugs is *mentioned* in a sentence, and if so, classify the interaction type. This task is treated as a sentence-level classification problem. Note that we are not deciding whether two drugs interact, but whether the target sentence says so.

A key objective of this lab is to expose you to different generations of NLP techniques and to help you understand their strengths and limitations in practice.

For both NERC and DDI tasks, you will experiment with three main approaches:

- Feature-based Machine Learning models, where linguistic and syntactic information is manually encoded and fed into classical classifiers.
- Neural Network models (e.g., LSTM or CNN architectures), where feature engineering is largely replaced by learned representations and end-to-end training.
- Large Language Models (LLMs), using prompt-based inference, few-shot learning, and lightweight fine-tuning.

The goal of this project is not to build the best possible system at any cost, but to understand how different NLP paradigms behave when applied to the same task under controlled conditions.

By working with all three paradigms on the same tasks and datasets, you will be expected to directly compare their performance, development effort, interpretability, and computational cost.

3 Learning Objectives

By the end of this laboratory project, you should be able to:

- Understand NLP pipelines for named entity recognition and relation extraction.
- Experiment with different modeling paradigms and analyze their impact on performance.
- Develop intuition about trade-offs between model complexity, development effort, computational cost, and results.
- Design and evaluate ML, neural, and LLM-based solutions using standard benchmarks.
- Produce clear experimental documentation and comparative analysis.

4 Methodology and Timeline

The laboratory project is organized into sequential goals that progressively introduce different approaches to the same underlying tasks. Each phase builds on the previous ones, allowing you to reuse knowledge, datasets, and evaluation tools.

For each task (NERC and DDI), a reference implementation is provided. You are expected to improve system performance by adjusting different system components.

A final comparative analysis is required, where students reflect on achieved performance, development effort, computational cost, and practical trade-offs across all approaches. Special attention is given to the fact that the task domain is narrow and specialized, making it an interesting case study for comparing traditional NLP pipelines against modern LLM-based solutions.

Laboratory sessions are complemented by group work outside class time. You are expected to continue experimenting between sessions.

Task 1 - NERC

- **System 1.1 – Sessions 1-2 - *NERC with Machine Learning.***

  A working feature-based Machine Learning approach to the NERC task is provided. Your goal is to understand the BIO sequence labeling setup, and improve the system performance by:

  - Experimenting with additional feature templates
  - Experimenting with different learning algorithms and hyperparameter settings

- **System 1.2 – Sessions 3-4 - *NERC with Neural Networks***

  A working a LSTM-based system is provided. Your goal is to understand the architecture of the model and the differences in the input encoding with respect to the ML system, and to improve system performance by:

  - Experimenting with input representations
  - Experimenting with network architecture variants (layers, sizes, dropouts, etc)
  - Tuning model hyperparameters

- **System 1.3 – Sessions 5-7 - *NERC with LLMs***

  A working LLM-based system is provided. Your goal is to understand how the prompt and example selection affects the results, and to improve system performance by:

  - Improving/adapting the prompts to overcome system errors.
  - Experimenting with few-shot training, including number of provided examples and used example selection criteria.
  - Experimenting with fine-tuning, including number of tuning examples and used example selection criteria.

Deliverables: At the end of Task 1, you will be required to **deliver a report** describing:

- NERC-ML performed experiments (features, algorithms, hyperparameters, ...). Effect of tried variations. Best obtained results.
- NERC-NN performed experiments (input, architecture, hyperparameters, ...). Effect of tried variations. Best obtained results.
- NERC-LLM performed experiments (prompt adaptation, example selection for few-shot, example selection for fine-tuning, fine-tuning hyperparameters...). Effect of tried variations. Best obtained results.
- Conclusions. Comparison of the different approaches in terms of required development effort, computational cost, explainability and system behaviour controllability, and resulting performance.

Task 2 - DDI

- **System 2.1 – Sessions 8-9 - *DDI with Machine Learning***

  A working feature-based Machine Learning approach to the DDI task is provided. Your goal is to understand the problem modeling as a sentence classifier, and improve the system performance by:

  - Experimenting with additional feature templates, with special attention to syntax-based features.
  - Experimenting with different learning algorithms and hyperparameter settings

- **System 2.2 – Session 10 - *DDI with Neural Networks***

  A working a CNN-based system is provided. Your goal is to understand the architecture of the model and the differences in the input encoding with respect to the ML system, and to improve system performance by:

  - Experimenting with network architecture variants (layers, sizes, dropouts, etc)
  - Tuning model hyperparameters

- **System 2.3 – Sessions 11-13 - *DDI with LLMs***

  A working LLM-based system is provided. Your goal is to understand how the prompt and example selection affects the results, and to improve system performance by:

  - Improving/adapting the prompts to overcome system errors.
  - Experimenting with few-shot training, including number of provided examples and used example selection criteria.
  - Experimenting with fine-tuning, including number of tuning examples and used example selection criteria.

Deliverables: At the end of Task 2, you will be required to **deliver a report** describing:

- DDI-ML performed experiments (features, algorithms, hyperparameters, ...). Effect of tried variations. Best obtained results.
- DDI-NN performed experiments (input, architecture, hyperparameters, ...). Effect of tried variations. Best obtained results.
- DDI-LLM performed experiments (prompt adaptation, example selection for few-shot, example selection for fine-tuning, fine-tuning hyperparameters...). Effect of tried variations. Best obtained results.
- Conclusions. Comparison of the different approaches in terms of required development effort, computational cost, explainability and system behaviour controllability, and resulting performance.