Teaching Modern NLP and LLMs at Kyiv School of Economics: A Practice-Oriented Course with Ukrainian Language Focus

Roman Kyslyi, Anton Bazdyrev
Kyiv School of Economics, Ukraine
rkyslyi@kse.org.ua, abazdyrev@kse.org.ua

Abstract

This paper describes a Natural Language Processing (NLP) course taught at Kyiv School of Economics. The course consists of 16 lectures, 5 practical assignments and focuses on modern large language models (LLMs) while preserving an introduction to classical NLP. Practical assignments are organized using Kaggle, where GPU support plays an important role in enabling students to work with complex models. A key feature of the course is the focus on Ukrainian in the practical assignments, contributing to the development of Ukrainian NLP expertise and community. The course is taught primarily in-person, but due to the ongoing war in Ukraine, also includes a full online participation option and additional weekly QnA sessions.

1 Introduction

Natural Language Processing has undergone rapid changes in recent years, largely driven by the success of transformer-based architectures and large language models (Vaswani et al., 2017). As a result, NLP courses must be updated frequently to reflect current practice. At the same time, students benefit from understanding classical NLP concepts such as pre-transformer tokenization, embeddings, vectorization, and evaluation, which remain relevant even in modern LLM-based systems (Jurafsky and Martin, 2023).

Another challenge in NLP education is language diversity. Many courses, even in non-English speaking countries, rely mostly on English datasets and models, which hides important issues related to morphology, data scarcity, and domain mismatch. Ukrainian is a mid-resource language where these issues are clearly visible. In recent years, Ukrainian NLP has been developing rapidly, with increased availability of datasets and models and growing community activity on open platforms such as Hugging Face (Wolf et al., 2020). We encourage our students to participate in annual Ukrainian Natural Language Processing (UNLP) conference (Romanyshyn, 2025) which is dedicated to the development of Ukrainian NLP and has grown substantially in recent years. Thus, actively using Ukrainian in NLP education is timely and important.

The NLP course at KSE was designed to address these challenges. It combines classical NLP foundations with modern LLM-based methods, uses Ukrainian as a primary language for practical tasks, and is adapted to the constraints of wartime teaching.

2 Course Context and Learning Format

The course is taught at KSE as a one-trimester course consisting of sixteen lectures and five practical assignments. Lectures are held in-person whenever possible and emphasize conceptual understanding, discussion, and system-level intuition. At the same time, students are able to attend lectures online if physical presence is not possible.

Due to constraints like air raid alerts, power outages, and other disruptions caused by the war, flexibility is essential. All lecture materials are distributed online, and a weekly evening QnA session is organized to support students who miss classes or need additional clarification. Similar hybrid and remote teaching adaptations have been reported as necessary in other crisis-affected educational contexts (Hodges et al., 2020).

3 Classical NLP as a Foundation

Although the course focuses strongly on LLMs, it begins with classical NLP methods. Students learn how NLP tasks are formulated, how text is represented using classical methods like BoW and TF–IDF (Salton and Buckley, 1988) models, and how results are evaluated using standard classification and generation metrics.

This foundation helps students better understand

Proceedings of the Seventh Workshop on Teaching Natural Language Processing (TeachNLP 2026), pages 84–87
March 29, 2026

modern LLM behavior. Issues such as tokenization, data imbalance, and metric limitations become easier to interpret when students have experience with simpler models. Similar motivations for integrating classical NLP into modern curricula are discussed in existing NLP teaching literature (Manning, 2022).

4 Practical Assignments and Kaggle Infrastructure

All practical assignments are organized using Kaggle. It provides a standardized environment for dataset distribution, submission tracking, and leaderboard-based evaluation. An important reason for this choice is GPU availability, which is crucial for fine-tuning transformer-like models and running LLM-based experiments within reasonable time limits in an educational setting (Wolf et al., 2020).

The five practical assignments cover the following topics (detailed description can be found in the Appendix A):

- Classical NLP pipelines and subword tokenization
- Sequence-to-sequence models for summarization or translation
- Supervised fine-tuning and alignment of large language models
- Retrieval-augmented generation systems (Lewis et al., 2020; Izacard and Grave, 2021)
- Agentic LLM workflows with tool usage

The assignment sequence shows a progression from classical NLP to modern LLM systems, following common design patterns in modern NLP pipelines. It is also consistent with usual Shared Tasks at UNLP conference, like fine-tuning LLMs for Ukrainian question answering (Romanyshyn et al., 2024) or detecting social media manipulation in Ukrainian (Kyslyi et al., 2025) which allow students to participate and smoothly join the research community.

5 Ukrainian NLP as a Core Component

An important aspect of the course is the usage of Ukrainian in practical assignments. Students work with Ukrainian texts in tasks such as classification, summarization, retrieval, and instruction tuning. This is motivated not only by pedagogical considerations, but also by the practical properties of Ukrainian as a mid-resource language. Training data availability and its morphological complexity make practical assignments more challenging, compared to English-centric tasks. Also, pretrained multilingual models frequently do not have the best quality for Ukrainian, which makes classical baselines and careful error analysis particularly valuable in teaching.

As a result, students are exposed not only to standard NLP workflows, but also to realistic research constraints typical for mid-resource languages. Working with Ukrainian data requires more careful model training and evaluation results analysis, enabling a deeper understanding of practical limitations in NLP systems.

6 Discussion

Several observations emerged during the course. Students are developing a deeper understanding of LLMs when classical NLP methods are introduced first, as this provides a better understanding of the NLP landscape. In particular, prior experience with tokenization and classical baselines helps students better interpret LLM failures related to context length, data imbalance, and evaluation metrics. Kaggle-based practical assignments enable realistic experimentation even in relatively large classes, while keeping the technical setup manageable for both students and instructors.

We also observed that using non-English for practical tasks encourages students to rely less on default pretrained models and more on careful dataset analysis and model adaptation.

A recurring discussion topic in the course concerns cross-lingual and cross-Slavic transfer. Students are encouraged to explore whether models pretrained on other Slavic languages can be adapted to Ukrainian via fine-tuning. While lexical and syntactic similarities exist, Ukrainian differs substantially from languages such as Polish or Slovak in morphology, vocabulary, and orthography. In practice, naive transfer often leads to degraded performance or subtle semantic errors, motivating careful evaluation and language-specific adaptation rather than assuming interchangeability between Slavic languages.

This helps students develop intuition about the limits of multilingual pretraining and the risks of assuming linguistic proximity implies model compatibility.

The hybrid teaching format, although initially forced by external constraints, has proven to be effective: the possibility to attend lectures online and the availability of weekly evening QnA sessions help maintain engagement and learning continuity despite disruptions.

7 Conclusion

This paper presented an NLP course at Kyiv School of Economics that highlights practical work with Ukrainian data. In our experience, mid-resource languages (like Ukrainian) are particularly effective for teaching NLP, as they reveal limitations and constraints more clearly than English-centric settings while remaining accessible for coursework. The course design demonstrates how contemporary NLP education can remain technically up to date while adapting to multilingual contexts and non-standard teaching conditions.

References

Charles Hodges, Stephanie Moore, Barb Lockee, Torrey Trust, and Aaron Bond. 2020. The difference between emergency remote teaching and online learning. *Educause Review*.

Gautier Izacard and Edouard Grave. 2021. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the ACL (EACL)*.

Daniel Jurafsky and James H. Martin. 2023. *Speech and Language Processing*. Prentice Hall. 3rd edition draft.

Roman Kyslyi, Nataliia Romanyshyn, and Volodymyr Sydorskyi. 2025. The unlp 2025 shared task on detecting social media manipulation. In *Proceedings of the Fourth Ukrainian Natural Language Processing Workshop (UNLP 2025)*, pages 105–111, Vienna, Austria (online). Association for Computational Linguistics.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Proceedings of the 34th Conference on Neural Information Processing Systems (NeurIPS)*.

Christopher D. Manning. 2022. Human language understanding and reasoning. Keynote at ACL 2022.

Mariana Romanyshyn, editor. 2025. *Proceedings of the Fourth Ukrainian Natural Language Processing Workshop (UNLP 2025)*. Association for Computational Linguistics, Vienna, Austria (online).

Mariana Romanyshyn, Volodymyr Sydorskyi, and Roman Kyslyi. 2024. Overview of the unlp 2024 shared task on ukrainian argument mining. In *Proceedings of the Third Ukrainian Natural Language Processing Workshop (UNLP 2024)*, pages 89–98, Dubai, United Arab Emirates (online). Association for Computational Linguistics.

Gerard Salton and Christopher Buckley. 1988. Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, 24(5):513–523.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st Conference on Neural Information Processing Systems (NeurIPS)*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, and 1 others. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*.

A Appendix A: Practical Assignments

The course includes five graded practical assignments. All assignments are implemented and evaluated using Kaggle.

Each assignment builds on previous ones, gradually increasing technical complexity and conceptual depth.

A.1 Classical NLP Pipelines and Tokenization

In the first assignment, students implement classical NLP pipelines for text classification. They experiment with different text representations, including bag-of-words and TF–IDF, combined with linear models such as logistic regression.

A key component of this assignment is tokenizer analysis. Students train subword tokenizers (BPE or WordPiece) on Ukrainian corpora and compare them with pretrained tokenizers. They analyze vocabulary size, token length distribution, and out-of-vocabulary behavior, and reflect on how tokenization choices affect downstream models.

A.2 Seq2Seq Models for Summarization or Translation

The second assignment introduces encoder–decoder transformer models. Students first apply pretrained models in zero-shot or few-shot settings and analyze common error patterns.

They then fine-tune a smaller model on a focused dataset, using Ukrainian text. Performance is evaluated using standard generative metrics (e.g., ROUGE or BLEU) as well as qualitative inspection. This assignment highlights the difference between prompt-based usage and task-specific training.

A.3 Supervised Fine-Tuning and Alignment of LLMs

In the third assignment, students perform supervised fine-tuning of a small or medium-sized LLM using parameter-efficient methods such as QLoRA. They compare the behavior of the base and fine-tuned models using both automatic metrics and manual evaluation.

An additional part of the assignment introduces preference-based alignment methods, such as Direct Preference Optimization (DPO) or GRPO. Using a small set of preference pairs, students observe how alignment affects helpfulness, safety, and instruction-following behavior, and discuss practical challenges such as data quality and training stability.

A.4 Retrieval-Augmented Generation

This assignment focuses on building a RAG system. Students first establish a baseline where an LLM answers questions without retrieval, identifying typical failure modes such as hallucination or missing context.

They then implement a full RAG pipeline including document chunking, vector indexing, retrieval, and context injection. In many cases, they observe that off-the-shelf multilingual embeddings perform poorly, motivating experiments with embedding fine-tuning or hybrid retrieval approaches.

A.5 Agentic LLM Workflows

The final assignment introduces agentic systems. Students design a simple agent that can plan actions, call predefined tools, and generate final answers based on intermediate results.

This assignment integrates concepts from earlier tasks, including retrieval, prompt design, and alignment. Students evaluate agent reliability, analyze failure cases, and reflect on how tool usage and control logic influence system behavior.

Appendix B: Topics covered

The course consists of sixteen lectures organized into four modules:

Module 1: Classical NLP and Text Representations

- Problem formulation, and classical pipelines (bag-of-words, TF–IDF etc.)
- Tokenization and subword modeling (BPE, WordPiece, unigram models)
- Neural word embeddings and distributional semantics (word2vec, fastText)

Module 2: Large Language Models

- Encoder and sequence-to-sequence models (T5, BART) for translation and summarization
- Decoder-only LLMs, supervised fine-tuning, and efficient inference
- Reinforcement learning foundations and RLHF for LLMs
- Preference-based alignment methods (DPO, GRPO)
- Vision–language and multimodal language models

Module 3: Retrieval-Augmented Generation

- Vector search and approximate nearest neighbour methods
- Sparse vs. dense retrieval and embedding fine-tuning
- End-to-end RAG pipelines and design patterns

Module 4: Advanced LLM Systems and Agentic AI

- MLOps for LLMs, including serving, quantization, and distillation
- Reasoning in LLMs and chain-of-thought prompting
- Prompt programming, DSPy, and lightweight tuning methods
- Agentic LLM systems with tool usage and planning