*Laboratory work - Spring 2025*

The main goal of laboratory work is to present the most important aspects of data science in practice and to teach you how to use key tools for a NLP engineer. We especially emphasize on self-paced work, raising standards related to development, replicability, reporting, research, visualizing, etc. Our goal is not to provide exact instructions or "make robots" out of participants of this course. Participants will need to try to navigate themselves among data, identify promising leads and extract as much information as possible from the data to present to the others (colleagues, instructors, companies or their superiors).

**Important links**

[Lab sessions course repository](https://github.com/UL-FRI-Zitnik/NLP-Course-Tutorials) (continuously updated, use weekly plan links for latest materials)

[Project report template](https://unilj-my.sharepoint.com/:u:/g/personal/slavkozitnik_fri1_uni-lj_si/EYACm1wPHMxDrtoQrOxUuIkBJhLFgAbkFPwsxrA_LeBXFw?e=s36wNg)

Books and other materials

[Speech and language processing](https://web.stanford.edu/~jurafsky/slp3/) (online draft)

[Python 3 Text Processing with NLTK 3 Cookbook](https://drive.google.com/file/d/1GiRyQHmu9Wh5b_vKoX2Ty4uUIncE2AEi/view?usp=sharing)

[Introduction to Data Science Handbook](https://fri-datascience.github.io/course_ids/handbook/)

[Razvoj slovenščine v digitalnem okolju](https://ebooks.uni-lj.si/ZalozbaUL/catalog/book/522) (February 2023)

Previous years NLP course materials

[NLP course 2024 project reports](https://unilj-my.sharepoint.com/:b:/g/personal/slavkozitnik_fri1_uni-lj_si/EXGeIa-Bg_xHvIRGSAHsocQB8gtT-T1DiFXlTud2Y-mpyw?e=YDMOcF)

Peer review

TBA

*\* Check Marks spreadsheet (sheet "Projects to review") to get list of repositories you need to review.*

**Weekly plan**

This plan is regularly updated.

*Lab sessions* are meant to discuss materials and your project ideas. Those proficient in some of the topics covered during the course are expected to help other students during the lab work or in online discussions. Such contributions will also be taken into account. During the lab sessions we will show some DEMOs based on which you will work on your projects. Based on your proposals / new ideas we can adapt the weekly plan and prepare additional materials.

All the lab session tutorials will be regularly updated in the [Github repository](https://github.com/UL-FRI-Zitnik/NLP-Course-Tutorials). During the lab sessions we will briefly present each week's topic and then mostly discuss your project ideas and work. You are expected to check/run notebooks before the lab sessions and then ask questions/discuss during the lab sessions. In the repository's README you can also find the recordings of each topic.

|  |  |  |
| --- | --- | --- |
| *Week* | *Description* | *Materials and links* |
| 17.2. - 21.2. | */* |  |
| 24.2. - 28.2. | [Lab work introduction](https://unilj-my.sharepoint.com/:b:/g/personal/slavkozitnik_fri1_uni-lj_si/EWrLt62pHnZAhQ6gqkDPGuMBrRGqOjxdrZrG8jw5Rl5NiA?e=8iHKlC)  Projects overview  Group work and projects application procedure  [Basic text processing](https://github.com/UL-FRI-Zitnik/NLP-Course-Tutorials/tree/master/01%20-%20Basic%20text%20processing)  [Slovene text processing](https://github.com/UL-FRI-Zitnik/NLP-Course-Tutorials/tree/master/02%20-%20Slovene%20text%20processing%20and%20rule-based%20systems) |  |
| 3.3. - 7.3. | Text clustering  Text classification  Language models, knowledge bases | [*Projects sign up form*](https://forms.office.com/e/gmKAN5WanB) *(****deadline Friday midnight****).*  [*Github classroom assignment*](https://classroom.github.com/a/kmDmdVbj) *(****deadline Friday midnight****, only one group member creates a team, exactly three members for a group!). Use* ***GROUP ID*** *from the projects sign up form as a team name.* |
| 10.3. - 14.3. | Neural networks introduction (TensorFlow, Keras)  Word embeddings & visualizations (offensive language)  RNNs vs. GRUs vs. LSTMs + examples  Multiple simple NN architectures example (Google Colab) | Wednesday’s lab session at 5pm cancelled due to <https://hackathon.si>. **You are invited to join!** |
| 17.3. - 21.3. | Introduction to PyTorch  PyTorch Lightning  SLING tutorial (setup, Singularity, SLURM)  **First submission (Friday, 23:59)** |  |
| 24.3. - 28.3. | *First submission defense (in person)*  *March 25: NVIDIA Day at UL FRI (*[*more info*](https://unilj-my.sharepoint.com/:b:/g/personal/slavkozitnik_fri1_uni-lj_si/Ec6Zzb4crFlLqRWjm4FG9REBQ4GVherFaE6rcdmnytbpyA?e=MpLVYy)*)* | |
| 31.3. - 4.4. | Transformers, BERT (custom task),  BERT (tagging, classification),  KeyBERT (keyword extraction), TopicBERT (topic modeling) |  |
| 7.4. - 11.4. | Generative and conversational AI |  |
| 14.4. - 18.4. | Prompting and efficiently Fine-Tuning a Large Language Model  Retrieval Augmented Generation (RAG) | [ARNES & SDI hackathon](https://hackathon.si/) in Portorož. You can get **up to additional 10%** for lab grade if you attend. |
| 21.4. - 25.4.  (Mon. holiday) | Graph neural networks for text processing |  |
| 28.4. - 2.5.  (Thu., Fri. holiday) | *No lab sessions*  **Second submission (Friday, 23:59)** | |
| 5.5. - 9.5. | *Second submission defense (in person)* | |
| 12.5. - 16.5. | *Consultations/Project work/Discussions*  *(Please attend the lab sessions and discuss your work and ideas!)* | |
| 19.5. - 23.5. | *Project work/Online discussions*  **Final submission deadline (Friday, 23:59)**  ***IMPORTANT: Put your repositories visibility to public before the deadline or shortly after!***  **Peer review submission deadline (TBD)**  Peer review link (each group will get an email with repositories to review)! | |
| 26.5. - 30.5. | *No organized lab sessions this week*  Final project presentations (TBD) | |

**Course obligations**

Please regularly check *Weekly plan* and course announcements for possible changes. You are expected to attend the sessions but you ***must*** attend the defense sessions. At the assignment defense dates at least one member of a group must be present, otherwise all need to provide their doctor’s justification. At the last assignment all members must be present and also need to understand all parts of the submitted solution.

All the work must be submitted using your Github project repository. Submission deadlines are indicated in the table above. Submission defenses will be held during the lab sessions.

Students must work in groups of **three members**! There can exist only one group of two members per project type. The distribution of work between members should be seen by commits within the repository.

|  |  |  |
| --- | --- | --- |
| *Obligation* | *Description* | *Final grade relevance* |
| Submission 1 | Project selection & simple corpus analysis  - Group (three members) selection  - Report containing Introduction, existing solutions/related work and initial ideas  - Proposed project dataset/corpus/data/...  - Well organized repository | *Pass* |
| Submission 2 | Initial implementation / baseline with results  - Updated Submission 1 parts  - Implemented at least one solution with analysis  - Future directions and ideas  - Well organized repository | *Pass* |
| Submission 3 | Final solution and report  - Final report incl. analyses and discussions  - Fully reproducible repository | *80%* |
| Peer review | Evaluate your peer group's work  - Each group will check final submissions of two other peer groups having the same topic | *20%* |
|  |  | *Total: 100%* |

**Grading criteria**

All the graded work is group work. All the work is graded following the scoring schema below. To successfully pass the laboratory work, you need to pass Submission 1 and 2, and gain 50% total from **each of** Submission 3 and Peer review.

Use PUBLIC GROUP ID for public communication regarding your group. GROUP ID is your internal id of a group for which marks will be publicly available.

*Scoring*

Scoring is done relative to achievements of all the participants in the course. Instructions will define the criteria the participants need to address and exactly fulfilling the instructions will result in score 8. All the other scores will be relative to the quality of the submitted work. The role of instructors is to find an appropriate clustering of all the works into 6 clusters. To better illustrate the scoring, schema will be as follows:

* 10 - exceptional: Extraordinary results, quality of work or extremely well structured and justified report. There might still be some very minor possibilities for improvement.

Repository is clear and runnable. Report is well organized, results are discussed, visualizations are added value to the text and well prepared. Apart from minimum criteria the group tried multiple of their novel ideas to approach the problem.

* 9 - very good: Above average knowledge presentation with some errors.

Same as above - it can be visible that a group had novel ideas but they got out of time or did not finish (polish) everything. The submission has multiple minor flaws.

* 8 - good: Submission of solid work, mainly addressing given instructions only.

Group implemented everything suggested by the minimum criteria but not investigated further (not found much related work, did not apply multiple other techniques, ...).

* 7 - superficial: Below average knowledge and submission of work with errors that show partial understanding.

Group implemented everything suggested by the minimum criteria but did not discuss results well, performed simple analyses only, ... Also the report is not well organized, and lacks data for reproducibility.

* 6 - sufficient: Minimum criteria addressed with some major errors and drawbacks.

Group was trying to implement minimum criteria (or part only) but their work has many minor flaws or a few major ones. The report also reflects their motivation.

* 5 - insufficient: Too much lack of knowledge, too many major errors or no work-effort could be drawn from the submitted work.

The group did not address one or more points of the minimum criteria and the report contains major flaws. It can be seen that the group did not invest enough time into the work.

**Final project preparation guidelines and peer review instructions**

Some major remarks that you should keep in mind out are the following regarding final submission:

* Comment on all the specifics of your algorithms that you use or have designed (i.e. features, hyperparameters, ...). In some cases it is useful to include an image instead of providing long descriptions.
* When you include graphs/images, they must be readable and provide additional insight (e.g. if there are lots of lines across the image or little space between them, it is not okay). When you report on results, keep them as much as possible in one table, so that a reader can compare different configurations. Also, it is useful to bold the best results.
* Both images and tables should be self-contained - i.e., together with the caption they need to provide enough information to the reader to understand it's meaning without reading text around.
* Keep your report concise and try to not submit a report longer than 4 pages + references + appendices. Also, make sure you follow the proposed template.
* Focus on reporting results and using sensible measures. Try to find examples where your algorithm works better and may not even work at all. Explain why and also justify the differences in approaches that you used. In case previous work exists for your dataset, put the best results of other researchers in your results table (even if your results are much lower).
* Your submitted work (repository and report) should be structured in a way that your colleagues would be able to understand and re-run everything. Include all dependencies for you projects:
  + In case you have used a non-public (or semi-public) dataset, do not include it in the repository, just put your contact data or protected link to download data/provide other instructions to retrieve data.
  + Datasets that are available elsewhere should be just linked in your report/repository. If you performed additional transformations on datasets, scripts for that should be available in the repository.
  + Some of you used models that take longer to train. You can include those models (maybe just the best one) in the repository or elsewhere and link it.
* Lastly, check that your repositories are publicly available before the peer review period starts! ***IMPORTANT***:
  + Include links to all dependencies/corpora or include them in the repository, so that anyone can check your work. Also include annotated data (if you manually prepared a corpus) or trained models (if training takes a long time).
  + For anyone that will review your work, it must be as simple as possible to run your code.

Peer review instructions:

1. Please find the projects you need to review (see link above).
2. Each group needs to review projects of the same topic they have chosen.
3. Submit your peer review scores in the Google Form (see link above).
4. You will get a score also for your grading, depending on how much (of course by some margin) your grading will be different from the assistant's grading.
5. Follow the scoring criteria as presented above and include feedback to your mark.

**Final project presentation instructions**

Each group will have **max. 3 minutes (STRICT)** to present their project. I will put your report to the projector and you will present it along with your report. I propose that you focus on specific interesting part that you will present (e.g., table, graph, figure, ...). The most important aspect to present is:

What is the "**take-away message**" of your work? This should be concrete and concise, so that anyone can understand (also a completely lay person).

See timetable above for time slots of your presentation. If you cannot attend, please write to me to get an alternative time slot.

**Project 6: Automatic generation of Slovenian traffic news for RTV Slovenija (Slavko):** This project aims to use an existing LLM, »fine-tune« it, leverage prompt engineering techniques to generate short traffic reports. You are given Excel data from promet.si portal and your goal is to generate regular and important traffic news that are read by the radio presenters at RTV Slovenija. You also need to take into account guidelines and instructions to form the news. Currently, they hire students to manually check and type reports that are read every 30 minutes.

Methodology

1. Literature Review: Conduct a thorough review of existing research and select appropriate LLMs for the task. Review and prepare an exploratory report on the data provided.
2. Initial solution: Try to solve the task initially only by using prompt engineering techniques.
3. Evaulation definition: Define (semi-)automatic evaluation criteria and implement it. Take the following into account: identification of important news, correct roads namings, correct filtering, text lengths and words, ...
4. LLM (Parameter-efficient) fine-tuning: Improve an existing LLM to perform the task automatically. Provide an interface to do an interactive test.
5. Evaluation and Performance Analysis: Assess the effectiveness of each technique by measuring improvements in model performance, using appropriate automatic (P, R, F1) and human evaluation metrics.

Recommended Literature

* Lab session materials
* RTV Slo data: [LINK](https://unilj-my.sharepoint.com/:f:/g/personal/slavkozitnik_fri1_uni-lj_si/El8B9xMfNERAgWI4I-sL9p0B191MmgeTk2Z19TiKRTjtcg?e=fagk0Y). The data consists of:
  + Promet.si input resources (*Podatki - PrometnoPorocilo\_2022\_2023\_2024.xlsx*).
  + RTV Slo news texts to be read through the radio stations (*Podatki - rtvslo.si*).
  + Additional instructions for the students that manually type news texts (*PROMET, osnove.docx*, *PROMET.docx*).

**Projekt 6: Samodejno generiranje slovenskih prometnih novic za RTV Slovenija (Slavko)**

Cilj tega projekta je uporaba obstoječega velikega jezikovnega modela (LLM), njegova prilagoditev ("fine-tuning") ter uporaba tehnik inženiringa pozivov (prompt engineering) za ustvarjanje kratkih prometnih poročil. Na voljo imate podatke v Excelovi obliki s portala promet.si, vaša naloga pa je ustvarjati redne in pomembne prometne novice, ki jih berejo radijski napovedovalci na RTV Slovenija. Pri tem morate upoštevati smernice in navodila za oblikovanje novic. Trenutno za to nalogo najemajo študente, ki ročno preverjajo in vnašajo poročila vsakih 30 minut.

Metodologija

1. Pregled literature – Izvedite temeljit pregled obstoječih raziskav in izberite ustrezne velike jezikovne modele (LLM) za to nalogo. Preglejte in pripravite raziskovalno poročilo o danih podatkih.
2. Začetna rešitev – Poskusite nalogo rešiti zgolj z uporabo tehnik inženiringa pozivov.
3. Določitev evalvacije – Določite (pol)avtomatska evalvacijska merila in jih implementirajte. Upoštevajte naslednje dejavnike: identifikacija pomembnih novic, pravilno poimenovanje cest, pravilno filtriranje, dolžina besedil itd.
4. Prilagoditev LLM (učinkovita uporaba parametrov) – Izboljšajte obstoječ LLM tako, da bo lahko samodejno opravljal nalogo. Razvijte vmesnik za interaktivno testiranje.
5. Evalvacija in analiza učinkovitosti – Ocenite uspešnost posamezne tehnike z merjenjem izboljšav modela s pomočjo ustreznih avtomatskih metrik (P, R, F1) in človeške evalvacije.

Priporočena literatura

* Gradiva iz vaj
* Podatki RTV Slovenija: LINK. Podatki vključujejo:
  + Vhodne vire s portala promet.si (datoteka *PrometnoPorocilo\_2022\_2023\_2024.xlsx*).
  + Tekste prometnih novic RTV Slovenija (*Podatki - rtvslo.si*).
  + Dodatna navodila za študente, ki ročno vnašajo prometne novice (*PROMET\_osnove.docx, PROMET.docx*).