194.093 NLP and IE — exercise description

TU Wien, 2023WS

The project exercise described in this document will allow you to get acquainted with all steps of solving a complex NLP task, starting with data preprocessing and simple baseline solutions and moving towards more complex approaches. All course credit will be awarded based on this project, therefore it is designed to keep you busy throughout the semester. Important dates and deadlines are summarized at the end of this document, in Section 4.

1 Summary

Task selection By the end of **Week 2** (October 15) you must **form groups of 4 and choose your preferred projec topics**. You may choose any of the tasks offered in Section 3 or any custom task that involves text classification and which has been approved by the exercise coordinator (see Section 3 for requirements and Section 5 for contact details). Registration of groups takes place via TUWEL, preferred topics can be selected using this form. Based on your selection your team will be assigned a mentor who will support you throughout the semester and evaluate each of your submissions. Your group will be added to a GitHub repository for pushing your submissions, details are in Section 2.

Milestone 1 By the end of **Week 5** (November 5) you shall have your core text datasets **preprocessed and stored in a standard format**. All code necessary for preprocessing must be pushed to your repository and briefly documented. More detail will be provided in the lecture on text processing (Week 2).

Milestone 2 By the end of **Week 9** (December 3) you shall implement **multiple baseline solutions** to your main text classification task. These should include both deep learning (DL) based methods such as those introduced in Weeks 5-6 but also non-DL models such as those shown in Week 3. Baselines can also include simple rule-based methods (e.g. keyword matching or regular expressions). Each baseline should be evaluated both quantitatively and qualitatively, more details will be provided in the lecture on text classification (Week 3)

Final solution Your final solution is due by the end of **January 28**. Final presentations will take place on **January 19** (a week after the final lecture), the week after that should be reserved for improvements based on feedback from the presentation. Your final submission should include all your code with documentation, a management summary (see Section 2), and your presentation slides.

Evaluation Your final grade will be determined by scores given on **the final solution** (50%), **the two milestones** (15% each), the **presentation**, and the **management summary** (10% each). Note that the milestone scores will be based on the state of your repository at the time of each milestone deadline. The score for your final project will be based on its originality, the quality of your analysis and discussion, the quality of your code, and on overall impression. You will receive individual scores and feedback for each of these aspects. Milestone 1 and Milestone 2 must each be completed with a minimum score of 35% by their respective deadlines to pass the course.

A note on expectations In the second half of the semester the lectures will introduce approaches to modeling linguistic structure and meaning, then provide an overview of approaches to some of the most common tasks in NLP, some of which may be applicable to your chosen topic. For your final solution you are expected to conceive and implement approaches that go beyond the standard baselines implemented in the first half of the semester. The value of these solutions may come not only from superior quantitative performance but also from better explainability, broader applicability (e.g. different domains, less data), simplicity, efficiency, etc. You are encouraged to approach your mentor to discuss your ideas and get feedback. Extensive optimization of the metaparameters of machine learning models for small quantitative gains will not be highly valued.

2 Additional instructions

Goals The topic descriptions in Section 3 provide many pointers and ideas for getting started, and indicate some challenges and questions that you can work on. You are not expected to address more than 1-2 of the challenges and questions listed, but the value of your project comes from your contributions to these (the implementation of standard methods with existing datasets can only satisfy Milestones 1 and 2). Quantitative performance of a solution is only one indicator of its value, based on the topic and the nature of your solution you may also need to consider aspects such as complexity, explainability, sustainability, risk of unintended bias, applicability (to multiple domains, datasets, or languages), etc.

Datasets and languages Each topic description makes some recommendations on datasets, but you are encouraged to find additional resources. Using datasets in languages other than English or German that are understood by members of your group is encouraged, and so is working on more than one language in the project. If you choose a language for which datasets are already available, consider using at least two of them in the project. You may also choose a language with no datasets, in this case your main challenge will be to find possible ways to bootstrap a solution and/or a dataset.

Evaluation Proper evaluation of methods, including your own, both quantitative (e.g. precision and recall) and qualitative (e.g. looking at the data), is essential. For some tasks and some datasets you cannot assume that higher figures mean better solutions. Some manual analysis of a system's output is usually necessary to understand its strengths and limitations. Topic descriptions may indicate task-specific challenges of evaluation.

Technical details After teams registered for topics they will receive instructions on how to create their project repository using GitHub Classroom. Teams should then push their solutions to this repository. The template repository will contain detailed instructions on how to structure your code and documentation, you can preview it here. Your solution should be implemented in **Python 3.7** or higher and should generally conform to **PEP8** guidelines. You should also observe **clean code** principles.

Management summary Your submission must be accompanied by a **2-page PDF document** that presents a summary of your solution — this is a **management summary**, so it should be written in a way that is easy to understand by top management, not NLP colleagues. The summary should contain an overview of the task, the challenges you faced, the external resources you used, the solution you implemented and its limitations, and possible next steps.

Final Presentation Each group will present the main results of their work to all other groups working on the same topic. The format is **20 minutes of presentation and 10 minutes of discussion** — we will be very strict with the timing, and stop the presentation at the 20 minute mark. **Each team member must present their own contributions to their project, so that they can be evaluated individually.** The presentation should be aimed at NLP colleagues, so highlight which approaches and techniques you used, which data you used, and the insights obtained. Presentation slides must be pushed to your project repository the day before the presentations. The schedule of presentations will be announced via TUWEL, please attend all presentations in your section.

3 Topics

Your group will work on ONE of the following topics. We will assign topics to groups based on your preferences, but we cannot guarantee that each group can work on their first choice. Use

the form in TUWEL to provide a list of three topics that you would like to work on, in order of preference. If your group would like to propose a topic that is not in the list, contact the exercise coordinator. The instructor listed for your chosen topic will be your point of contact in case of questions, you are encouraged to consult them (see Section 5 for contact details).

Topic 1: Stress Detection

Instructor Varvara Arzt

Overview The goal of this task is the classification of short utterances on social media (Reddit posts) to determine whether they contain stress.

Resources The Dreaddit dataset used for this task is available here. Please read the paper published about the dataset (Turcan and McKeown, 2019).

Questions and challenges

- Thoroughly study the Dreaddit dataset and the way how it was created.
- Try to train a feature-based dicriminative classifier using additional metadata derived from Reddit such as posting time or number of comments (*social_timestamp* and *social_num_comments* in .csv files containing the dataset) as well as LIWC information provided extra by the dataset authors. For more detail on the Linguistic Inquiry and Word Count (LIWC) see Pennebaker et al. (2015) as well as the initial paper that describes Dreaddit. The use of which features result in better model performance?
- Compare the results of a feature-based classifier with the results produced by a deep-learning model of your choise (e.g. a BERT-based model).
- Turcan and McKeown (2019) also describes both experiments with traditional supervised models like decision trees and neural models. Compare your results with those presented in paper.
- How can you make your classifier's decisions explainable to users?

Topic 2: Detection of Online Sexism in English Text

Instructor Varvara Arzt

Overview The goal of this task is a binary classification of short utterances on social media (*Reddit* comments and *Gab* posts) to determine whether they are sexist (the dataset contains a more fine-grained sexism detection but you are supposed to work only with labels *sexist*/*not sexist*, which are included in a *label_sexist* column in .csv files with the dataset). Therefore you can just ignore the columns *label_category* and *label_vector*.

The EDOS dataset used for this task is partially annotated. You can decide to focus on the annotated part of the dataset and build a model that can predict discrimination, or you can work with the full dataset and predict each utterance's label.

Resources The dataset used for this task is available on GitHub. Please read the paper published about the dataset (Kirk et al., 2023).

Disclaimer This dataset contains language that might be disturbing for some people. If you start to feel uncomfortable with working with the data, immediately stop doing so and contact your instructors, together we can find a solution on how to continue working on the exercise.

Questions and challenges

- Perform experiments with several models from a family of BERT-based models (e.g. DeBERTa, RoBERTa, HateBERT, and DistilBERT)
- Can you see patterns that always correspond with sexist content?
- After performing error analysis try to find out whether there is a particular pattern that results in misclassified instances
- How can you make your classifier's decisions explainable to users?
- **Extra question (optional)**: if you are interested in LLMs, you can also think in the direction of using models like GPT-3 or LLaMA for the task of detecting online sexism.

Topic 3: Detection of Toxicity in Online Comments

Instructor Pia Pachinger, Gábor Recski

Overview The goal of this task is a binary classification of online comments to determine whether they are toxic (*toxic* ≈ likely to make someone leave a discussion or give up on sharing their opinion).

Resources The students of the course Advanced Information Retrieval 2023 did a great job with annotating parts of the data. Write to pia.pachinger@tuwien.ac.at to get access to the data. There are two datasets available:

- **English:** The English dataset comprises 1600 comments.
- **German:** The German dataset comprises 4500 comments.

Disclaimer These datasets, especially the English dataset, contain language that might be disturbing for some people. If you start to feel uncomfortable with working with the data, immediately stop doing so and contact your instructors, together we can find a solution on how to continue working on the exercise.

Questions and challenges

- Perform experiments with BERT-based models, e.g.:
  - **English:** DeBERTa, RoBERTa, HateBERT, and DistilBERT. Here is an example of a shared task in which these models were used (look at task A and exchange *sexism* with *toxicity*).
  - **German:** GBERT (base or large), a German language model based on BERT, or GELECTRA (base or large), a German language model based on ELECTRA. Here is an example of a shared task for which these models were used. Here is a more concrete example of their usage for the shared task, you can keep your solution simpler.
- Use the additional data in the dataset to make your classifier more explainable / perform better, for example:
  - The targets of toxic language were annotated in the comments. (For example, if someone commented *Bert is a moron.*, *Bert* would be annotated as the target of this toxic comment.)
  - Vulgarities of toxic and non-toxic comments were annotated in the text
- Can you spot errors in the data?

- After performing error analysis try to find out whether there is a particular pattern that results in misclassified instances
- How can you make your classifier's decisions explainable to users?
- **More advanced**: If you are interested in LLMs, you can also think in the direction of using models like LLaMA for the task of detecting online toxicity.

Topic 4: Relation Extraction

Instructor Varvara Arzt

Overview Relation extraction (RE) is the task of extracting semantic relationships between entities from a text. These relationships occur between two or more entities and are defined by certain semantic categories (e.g. Destination, Component, Employed by, Founded by, etc.). Entities usually fall into certain types (e.g. Organization, Person, Drug type, Location, etc.). The task is to build a classifier that learns to predict the relationship between entities. RE task usually aims to extract triples of a form <e1><relation type><e2>, where e1 and e2 are often defined as head and tail entities. Let's have an example sentence with two entities as relation candidates:

> **Elevation Partners**, the $1.9 billion private equity group that was founded by **Roger McNamee**.

Typically in RE tasks, two entities (in our case, *Elevation Partners* and *Roger McNamee*) and usually their types (COMPANY, PERSON) are given in a context (e.g. in a sentence), and the task is to classify the *relation* that the two entity holds (if there is any). For this example, the correct label would be *founded_by*.

Resources

- TACRED dataset (Zhang et al., 2017): data will be provided (under LDC User Agreement for Non-Members license)
- TACRED (Zhang et al., 2017) vs. TACREV (Alt, Gabryszak, and Hennig, 2020): for details on TACREV see the corresponding GitHub repository; TACRED dataset will be provided (under LDC User Agreement for Non-Members license)
- DocRED (Yao et al., 2019): data can be found here
- Biographical (Plum et al., 2022): data will be provided (under GPL-3.0 license)

Questions and challenges

General Questions relevant for all RE datasets listed above

- Thoroughly study the dataset you have chosen. How was the dataset created? Which data have been used for it? Which potential bias do they include? In which form text and labels are provided. What is the average length of text utterances in a dataset you have chosen? What additional information is also provided in a dataset (like evidence information in DocRED or Stanford Universal Dependencies Parser results in TACRED). How could one use this extra information (maybe in traditional ML algorithms)?
- RE differs from classical classification tasks in that information about the relation candidates (the two entities in question) also needs to be modeled. How would you construct such a machine learning model for a RE task?
- Sentence-level vs. document-level RE: what are the pros and cons of both approaches? Answer this question in reference to a dataset(s) you have chosen to work with.

- What would be your strategy of marking entities (head and tail entity) in a triplet <e1><relation type><e2> when training a deep learning model? Does it influence the performance of a model?
- How stable is your model's performance if you make small data perturbations such as adding a negation in a sentence in a test set?
- After performing error analysis find out which relation types tend to result misclassifications. Are there ambiguous relations that result it? Do you find a large fraction of noisy instances misclassified by the crowdworkers in the initial dataset? Can you identify clues contained in a dataset that can be exploited by models?
- **Extra question (optional)**: if you are interested in LLMs, you can also think in the direction of using models like GPT-3 or LLaMA for the RE task. For some inspiration you can check Wadhwa, Amir, and Wallace (2023)

DocRED

- How did the authors of the DocRED dataset obtain the entities contained in text utterances? What are possible bias of such an approach?
- What are the largest challenges in detecting cross-sentence relations? Please reinforce your remarks with results you get after training a RE classifier.

TACRED vs. TACREV

TACREV is a revisited version of a small part of the TACRED dataset (960 revisited instances in a dev set and 1,610 revisited instances in a test set). It aims to correct the errors produced by the Amazon Mechanical Turk crowdworkers while annotating the TACRED data.

- Map the revisited instances to the instances in the initial TACRED dataset and replace the noisy TACRED labels with those provided in TACREV (mapping can be done based on the id of the instances).
- Perform the same experiments with the initial TACRED dataset and TACREV. How much does the revisited version of TACRED contribute to a performance improvement of a RE classification model? Compare your results with the results provided in Zhang et al. (2017) that depicts experiments on initial TACRED and Alt, Gabryszak, and Hennig (2020) that depicts experiments performed on the revisited version of TACRED.

Biographical

- Compare your experiments with those described in Plum et al. (2022). Another possibility would also be to reproduce experiments described in this paper, i.e. training a BERT-based classifier.
- **Advanced question**: look at the list of labels in the Biographical dataset and compare them to a list of labels in the TACRED dataset. Would you add some extra labels to the Biographical dataset? Would these additional labels probably solve the problem with some miclassified instances?

Topic 5: Explainable Relation Extraction

Instructors Ádám Kovács, Gábor Recski

Overview Many popular NLP tasks, including RE, currently utilize state-of-the-art solutions that capture text meaning by leveraging neural language models based on the Transformer architecture (Vaswani et al., 2017), such as BERT (Devlin et al., 2019). Although these models achieve state-of-the-art scores on benchmarks, their inner workings often remain opaque, leading us to treat them as black boxes. An interesting research question would be to implement transparent, or "white-box," solutions using semantic graphs and interpretable graph patterns. This would enable a comparison of advantages and disadvantages against state-of-the-art BERT and LLM (Large Language Models like GPT-4) models in terms of performance, cost, speed, and more.

Resources

- Generic relation extraction datasets, e.g., the Semeval 2010 dataset (Hendrickx et al., 2010) and the TACRED dataset (Zhang et al., 2017).
- Domain-specific relation extraction on medical data:
  - Datasets such as the CrowdTruth (Dumitrache, Aroyo, and Welty, 2018) and the Food-Disease (Cenikj, Eftimov, and Koroušić Seljak, 2021). In both tasks, the relation to be classified is *cause* or *treat* between drugs and foods.
  - Other medical relation extraction resources, like the BLUE benchmark datasets: `DDI` (Herrero-Zazo et al., 2013), `ChemProt` (Taboureau et al., 2011), and the `i2b2 2010 shared task` (Uzuner et al., 2011).

Questions and Challenges Beyond creating machine learning or deep learning baselines for the RE task, students in this topic should develop a white-box solution. Tools like the POTATO library can be employed for extracting and crafting graph patterns for text classification, or spaCy for building patterns on dependency trees. A key inquiry is to assess the comparative performance of these white-box methods against deep learning-based systems. Additionally, students can evaluate different semantic parsers for the RE task, analyzing their respective strengths and weaknesses. Another challenge would be to design a solution based on an LLM, such as GPT-4, and measure its performance metrics, including aspects like cost. For implementing an LLM solution, spacy-llm is a recommended tool.

Topic X: Bring your own topic!

You are encouraged to propose your own topic! Please note the following criteria:

- the topic should include a text classification task at its core and there should be some annotated training data available for this task, otherwise milestones 1 and 2 cannot be completed. If you are unsure whether your topic is suitable, we are happy to advise you
- you are still required to work in teams of 4, so you should assemble a team to work on the project (if necessary you can also bring in external members who are not registered for the course)
- you should contact the exercise coordinator (Gábor Recski) about your topic proposal, we can discuss your ideas and recommend 1-2 instructors who can act as your mentors

4 List of Deadlines

06.10.2023 — Exercise and topics introduced

13.10.2023 — Milestone 1 introduced

15.10.2023, 23:55 — All group members must be registered for their project group in TUWEL and the group must fill out the topic selection form

20.10.2023 — Milestone 2 introduced

05.11.2023, 23:55 — Deadline for pushing Milestone 1 to GitHub

03.12.2023, 23:55 — Deadline for pushing Milestone 2 to GitHub

15.12.2023, 9-13h — Review meetings

18.1.2024, 23:55 — Deadline for pushing your presentation material to GitHub

19.1.2024 — Final presentations

26.1.2024, 23:55 — Deadline for pushing your final submission to GitHub

5 Contact

Administrative questions should be directed to the exercise coordinator, Gábor Recski.

Name	Email	Office hours
Varvara Arzt	varvara.arzt@tuwien.ac.at	see https://tiss.tuwien.ac.at/person/314093
Ádám Kovács	adam.kovacs@tuwien.ac.at	by appointment
Gábor Recski	gabor.recski@tuwien.ac.at	see https://tiss.tuwien.ac.at/person/336863

References

[1] Christoph Alt, Aleksandra Gabryszak, and Leonhard Hennig. "TACRED Revisited: A Thorough Evaluation of the TACRED Relation Extraction Task". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, July 2020, pp. 1558–1569. DOI: 10.18653/v1/2020.acl-main.142. URL: https://aclanthology.org/2020.acl-main.142.

[2] Gjorgjina Cenikj, Tome Eftimov, and Barbara Koroušić Seljak. "SAFFRON: tranSfer leArning For Food-disease RelatiOn extractioN". In: *Proceedings of the 20th Workshop on Biomedical Language Processing*. Online: Association for Computational Linguistics, June 2021, pp. 30–40. DOI: 10.18653/v1/2021.bionlp-1.4. URL: https://aclanthology.org/2021.bionlp-1.4.

[3] Jacob Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *Proc. of NAACL*. Minneapolis, Minnesota: Association for Computational Linguistics, 2019, pp. 4171–4186. DOI: 10.18653/v1/N19-1423. URL: https://aclanthology.org/N19-1423.

[4] Anca Dumitrache, Lora Aroyo, and Chris Welty. "Crowdsourcing Ground Truth for Medical Relation Extraction". In: *ACM Transactions on Interactive Intelligent Systems* 8.2 (2018), 1–20. ISSN: 2160-6463. DOI: 10.1145/3152889. URL: http://dx.doi.org/10.1145/3152889.

[5] Iris Hendrickx et al. "SemEval-2010 Task 8: Multi-Way Classification of Semantic Relations between Pairs of Nominals". In: *Proceedings of the 5th International Workshop on Semantic Evaluation*. Uppsala, Sweden: Association for Computational Linguistics, July 2010, pp. 33–38. URL: https://aclanthology.org/S10-1006.

[6] María Herrero-Zazo et al. "The DDI corpus: An annotated corpus with pharmacological substances and drug–drug interactions". In: *Journal of Biomedical Informatics* 46.5 (2013), pp. 914–920. ISSN: 1532-0464. DOI: https://doi.org/10.1016/j.jbi.2013.07.011. URL: https://www.sciencedirect.com/science/article/pii/S1532046413001123.

[7] Hannah Kirk et al. "SemEval-2023 Task 10: Explainable Detection of Online Sexism". In: *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*. Toronto, Canada: Association for Computational Linguistics, July 2023, pp. 2193–2210. DOI: 10.18653/v1/2023.semeval-1.305. URL: https://aclanthology.org/2023.semeval-1.305.

[8] James W. Pennebaker et al. "The Development and Psychometric Properties of LIWC2015". In: 2015. URL: https://api.semanticscholar.org/CorpusID:151038946.

[9] Alistair Plum et al. *Biographical: A Semi-Supervised Relation Extraction Dataset*. 2022. arXiv: 2205.00806 [cs.IR].

[10] O. Taboureau et al. "ChemProt: a disease chemical biology database". In: *Nucleic Acids Res* 39.Database issue (2011), pp. D367–372.

[11] Elsbeth Turcan and Kathy McKeown. "Dreaddit: A Reddit Dataset for Stress Analysis in Social Media". In: *Proceedings of the Tenth International Workshop on Health Text Mining and Information Analysis (LOUHI 2019)*. Hong Kong: Association for Computational Linguistics, Nov. 2019, pp. 97–107. DOI: 10.18653/v1/D19-6213. URL: https://aclanthology.org/D19-6213.

[12] Ö. Uzuner et al. "2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text". In: *J Am Med Inform Assoc* 18.5 (2011), pp. 552–556.

[13] Ashish Vaswani et al. "Attention is All you Need". In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon et al. Long Beach, CA, USA: Curran Associates, Inc., 2017, pp. 5998–6008. arXiv: 1706.03762 [cs.CL]. URL: http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf.

[14] Somin Wadhwa, Silvio Amir, and Byron C. Wallace. *Revisiting Relation Extraction in the era of Large Language Models*. 2023. arXiv: 2305.05003 [cs.CL].

[15] Yuan Yao et al. "DocRED: A Large-Scale Document-Level Relation Extraction Dataset". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, July 2019, pp. 764–777. DOI: 10.18653/v1/P19-1074. URL: https://aclanthology.org/P19-1074.

[16] Yuhao Zhang et al. "Position-aware Attention and Supervised Data Improve Slot Filling". In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, Sept. 2017, pp. 35–45. DOI: 10.18653/v1/D17-1004. URL: https://aclanthology.org/D17-1004.