194.093 NLP and IE — exercise description

1 Introduction

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 in the first weeks of the course and moving towards more complex approaches by the second half of the semester. Important dates and deadlines are summarized at the end of this document, in Section 8.

2 Overview

In the first weeks of the semester you are asked to **form groups of 4 and choose a task** from Section 5, based on which one or two course instructors will become your group's **mentor(s)**, they will support you throughout the semester and evaluate each of your submissions. You are also welcome to bring your own topic, see Section 5.5 for details. Your group will be added to a GitHub repository for pushing your submissions, details are in Section 4.

Each task will involve the use of 2-3 datasets, these have to be properly preprocessed, often using some of the methods described in **Lecture 1 on text processing**. The next step is the implementation of some simple, standard solutions, these are often called **baselines**. In **Lecture 2 on text classification** some simple, task-independent approaches will be introduced, and **Milestone 1** is the implementation of at least one simple baseline for your chosen task on at least one dataset, plus a very brief analysis of its strengths and weaknesses (not more than 1-3 paragraphs of text). This should be submitted by the end of **Week 6**, in the form of clean and documented Python code pushed to your project repository. Deep learning approaches in NLP will be introduced in **Lectures 3-5**, after this you will be able to train and evaluate a neural network architecture appropriate for your chosen task, this is **Milestone 2** and should be submitted by the end of **Week 9**, also in the form of clean and documented Python code.

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, any of which may be applicable to your chosen topic. Groups 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, from broader applicability (e.g. different domains, less data), simplicity, etc. You are encouraged to approach your mentor to discuss your ideas and get feedback. The final results must be communicated in both a 30 minute presentation (where all team members present their own contributions) and a 2-page management summary, and should be accompanied by a clean and readable software repository. For general and topic-specific instructions, see Sections 3 and 5, respectively. In the last week of the course each group will present their project, and should have submitted their software and report, all of which will be evaluated separately. Presentations will take place on **Week 14**, the deadline for the final submission is on Week 15, see Section 6 for detailed instructions and Section 7 on evaluation principles.

3 General instructions

3.1 Goals

The topic descriptions in Section 5 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.

3.2 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.

3.3 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.

4 Technical

4.1 Version control

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: https://github.com/tuw-nlp-ie/project-2022WS

4.2 Coding guidelines

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.

5 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 instructors. Read carefully both the general instructions in Section 3 and those specific to your chosen task below. The instructor listed for your chosen topic will be your point of contact in case of questions, you are encouraged to consult them. E-mail addresses are listed below:

Kinga Gémes kinga.gemes@tuwien.ac.at

Ádám Kovács adam.kovacs@tuwien.ac.at

Gábor Recski gabor.recski@tuwien.ac.at

5.1 Topic 1: Detection of targeted offensive text in rationale annotated text

Instructor Kinga Gémes, Gábor Recski

Overview The goal of this task is the classification of short utterances on social media (e.g. tweets, facebook comments, etc.) to determine whether they are offensive. The dataset used for this task also contains rationales, which could be used during the tasks. Choose a target group, that you want to focus on, and create models, that recognize text targeting the chosen group of people.

Resources The dataset used for this task is available on GitHub. Please read the paper published about the dataset Mathew et al., 2021.

Questions and challenges

- The dataset contains each annotator's opinion. How would you determine that a specific text is targeting a specific group if the annotators don't agree?
- How could you use the provided rationales?
- How can you make your classifier's decisions explainable to users? What are the limitations of the explanations you can provide?

5.2 Topic 2: Detection of offensive text in German text

Instructor Kinga Gémes, Gábor Recski

Overview The goal of this task is the classification of `Der Standard` comments to determine whether they are discriminative. The 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 comment's positive/negative vote.

Note Please only choose this topic if most people in your group speak or understand German.

Resources The dataset used for this task is available here or here. Please read the paper published about the dataset Schabus, Skowron, and Trapp, 2017.

Questions and challenges

- How consistent is the labeling across the dataset? Can you see patterns that always correspond with discriminative/negatively voted text?
- Is there a correlation between negatively voted texts and discriminative texts?
- How can you make your classifier's decisions explainable to users? What are the limitations of the explanations you can provide?

5.3 Topic 3: Relation Extraction

Instructor Ádám Kovács, Gábor Recski

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. 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

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

Questions and challenges

- 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?
- How would you leverage graphs (e.g. Universal Dependency trees) into your solution (idea: use paths between the entities as features)?
- **Advanced question:** In many popular NLP tasks (also in RE), the state-of-the-art solutions usually capture the meaning of the text by leveraging neural language models that are based on the Transformer architecture Vaswani et al., 2017 (e.g. BERT Devlin et al., 2019. While achieving state-of-the-art scores on benchmarks, these solutions are usually hard to interpret, and we treat them as black-box. An interesting research question would be to develop a white-box solution using semantic graphs and interpretable graph patterns. The POTATO library provides tools for extracting and developing graph patterns for text-classification tasks. In this task, the student could also use and compare different semantic parsers and how they fare against each other on the problem.
- **Advanced question:** The CrowdTruth and the FoodDisease datasets contain the same labels and similar entity types. How do modern neural based models (e.g. BERT (Devlin et al., 2018)) transfer their knowledge between the datasets? Do rule-based models transfer better?

5.4 Topic 4: Attribute extraction from building regulations

Instructor Eszter Iklódi, Gábor Recski

Overview As part of the BRISE project, 250 text documents of the Zoning Plan (Flächenwidmungsplan) of the City of Vienna have been annotated for the attributes they regulate. For example, the sentence *Der oberste Abschluss der zur Errichtung gelangenden Gebäude darf 15 m nicht überschreiten.* is annotated with the attribute `GebaeudeHoeheMax`. The documents are also annotated for the values and types of attributes as well as the rule structure, e.g. that the value of `GebaeudeHoeheMax` is `15m`, the type is `content` and the modality of the rule is `obligation`. The main task is the classification of sentences based on which attributes they mention. Possible next steps involve choosing some frequent attributes and also extracting their values and types from the text, or extracting the modality of the rules.

Note German language knowledge is useful for this topic as the dataset is in German.

Resources

- All relevant information and code regarding the annotation process and first experiments is available in the brise-plandok repository.
- A detailed description about the data is available here.
- The dataset itself divided into train-valid-test sets can be found here.
- The annotation guidelines are available here.
- Code and results of baseline ML experiments on the attribute extraction task are available here.
- You can find the code and results of our rule-base attribute extraction system here.

Questions and challenges

- Analyze the performance of your classifiers: which of the most frequent labels are also hard to detect?
- Compare your ML-based solutions to the existing rule-based system for the most frequent labels. What are the advantages of each? Which one would you choose? Would you have a way of combining them?
- For some attributes it is straightforward to also get the values and types, using simple patterns. Which ones are more difficult? How would you approach them?

5.5 Topic 5: Bring your own topic!

You are encouraged to propose your own topic! If you are interested, 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 (Gabor Recski) about your topic proposal, we can discuss your ideas and recommend 1-2 instructors who can act as your mentors

6 Submission, report, and final presentation

Submission: All material must be submitted by pushing to the project repository. Your final submission must contain clean, well-documented code as well as a final report and your presentation slides (see below for details).

Report: 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 a short discussion of where you now stand.

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.

7 Evaluation

The final mark will be based on the submitted code and report as well as the presentation. Milestone 1 and Milestone 2 must each be completed with a minimum score of 35% by their respective deadlines to pass the course.

The final mark is calculated from the following components:

- 15% for Milestone 1
- 15% for Milestone 2
- 50% for the final solution
- 10% for the presentation
- 10% for the management summary

Note that about 50 hours per person is foreseen for this exercise, around two-thirds of the time foreseen for the course (75 hours). This means that everyone should work more than a standard (40 hour) week on this exercise, so four weeks effort for a group of four. The evaluation will be based on the expectation of a manager (an NLP expert) assigning such a task to a team of four junior NLP engineers for a week. Note that this expectation is not met by submitting an overly long Jupyter notebook — you need to demonstrate that:

- You have approached the analysis in a logical and structured way.
- You are aware of the solutions already available for the problem, and show how your solution builds on them (note that you don't need a comprehensive state-of-the-art analysis).
- You have conducted experiments to show the effectiveness of your approach. Make sure you justify your choice of metrics.
- You have learned some new NLP tools and techniques.

Overly long notebooks with little substance will be penalised. If mature software already exists to solve your problem, it is not sufficient to simply submit this software as the solution. You should try and improve on the solution, or implement the solutions for another language. Black Box solutions are frowned upon — you should be able to explain to your manager how the model works and what its limitations are, in particular what it gets wrong and why. We suggest that you avoid complex neural network approaches for this exercise. Spending all of your time tuning the parameters of a complex model will not be highly evaluated.

8 List of Deadlines

Here is a list of the deadlines and what should be done by each deadline:

14.10.2022 — Exercise and topics introduced

21.10.2022 — Milestone 1 introduced

23.10.2022, 23:55 — All group members must be registered for their project group in TUWEL and the group must fill out the topic selection form

10.11.2022, 23:55 — Deadline for pushing Milestone 1 to GitHub

11.11.2022 — Milestone 2 introduced

01.12.2022, 23:55 — Deadline for pushing Milestone 2 to GitHub

19.1.2022, 23:55 — Deadline for pushing your presentation material to GitHub

20.1.2023 — Final presentations from all groups (in person)

26.1.2022, 23:55 — Deadline for pushing your final submission to GitHub

9 Office hours

Allan Hanbury Thursdays, 13:00-14:00
(see changes on this TISS page: https://tiss.tuwien.ac.at/person/48222).

Gábor Recski Tuesdays, 15:00-16:00
(see changes on this TISS page: https://tiss.tuwien.ac.at/person/336863).

References

[1] 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.

[2] Jacob Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". Version 1. In: *arXiv preprint arXiv:1810.04805* (Oct. 11, 2018). arXiv: 1810.04805v1 [cs.CL]. URL: http://arxiv.org/abs/1810.04805v1.

[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] Binny Mathew et al. "HateXplain: A Benchmark Dataset for Explainable Hate Speech Detection". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35. 17. 2021, pp. 14867–14875.

[8] Dietmar Schabus, Marcin Skowron, and Martin Trapp. "One Million Posts: A Data Set of German Online Discussions". In: *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. Tokyo, Japan, Aug. 2017, pp. 1241–1244. DOI: 10.1145/3077136.3080711.

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

[10] 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.

[11] 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.

[12] 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.