

Final Project Guidelines

*Prepared by Brielen Madureira, Jana Götze, David Schlangen (based on the versions from previous years, starting with Tatjana Scheffler's guidelines from 18/19).

1. Overview

Task. The grade of the course is determined based on a final project. Your task is to carry out a programming project using and extending the concepts covered in class. The final project will be a group effort. This will enable you to attempt a more ambitious project, and to make use of diverse backgrounds and abilities. To ensure a fair assessment, two individual papers are also part of the grade (a pre-project proposal paper and a post-project report).

You can select one of the following [SemEval 2023](#) subtasks:

A) Task 1: V-WSD: Visual Word Sense Disambiguation [website](#): Vision & language task. Given a word and some limited textual context, the task is to select among a set of candidate images the one which corresponds to the intended meaning of the target word.

B) Task 2. MultiCoNER II: Multilingual Complex Named Entity Recognition, [website](#): Sequence tagging problem where the task is to predict fine-grained types of named entities in a text document. The task includes data in 12 languages and you can choose any language to work with. You have to fill out a form to access the dataset.

C) Task 5: Clickbait Spoiling, [Subtask 1](#): Multiclass classification. Given a post, the task is to classify the spoiler type that the clickbait post warrants (either "phrase", "passage", "multi").

D) Task 6: LegalEval: Understanding Legal Texts, [Subtask A](#): Sequence classification. Given a document, the task is to automatically segment documents into semantically coherent text segments, and each such segment is assigned a label such as a preamble, fact, ratio, arguments, etc.

E) Task 6: LegalEval: Understanding Legal Texts, [Subtask B](#): Sequence tagging. Given a legal document, the task is to identify named entities such as "court", "petitioner", "respondent", etc.

F) Task 3: Detecting the Category, the Framing, and the Persuasion Techniques in Online News in a Multilingual Setup, [Subtask 1](#): Multiclass classification. Given a news article, determine whether it is an opinion piece, aims at objective news reporting, or is a satire piece.

Read the task description carefully. Check the datasets and the evaluation metrics. If the test split has not been released yet, you can create new splits from the available data, as long as you document well how you did it.

We have selected some tasks that allow you to access the dataset without registering to the competition. A few of the tasks do require personal information in order to access the data where you have to fill out a form on their websites. It is your choice if you want to do it or not, the priority is to opt for the open ones.

Remember that the goal here is to help you deepen your understanding of what we've done in class, and to document what you have understood by applying it to your selected problem. Your goal is *not* to beat the state of the art in anything. What we want to see is that you have gained some understanding of the tools that you've seen and can frame a given problem in such a way that you can tackle it with one of these

tools. What is important is the quality and systematicity of the work (selection of dataset, framing of question, execution of the experiment, analysis of the results), not necessarily the raw numerical evaluation score.

Scope. In terms of work load, the programming part of the project should correspond to approximately 1.5 x one of the course assignments. Please plan enough time. (And don't be *too* ambitious.) Working in groups will not decrease the overall time needed, just extend the scope of what you can do (slightly; also remember that group coordination costs time and effort, as well).

Grading. The grade takes the following criteria into account: planning paper (5%), project presentation (5%), project implementation (40%), and project report (50%). All the four components are mandatory in order to pass the exam.

2. Group work

Please work in groups of 2-3 students (preferably 3). Make some effort to create groups that include students from diverse academic backgrounds (i.e., don't just work with your friend from undergrad!). You will need much more than just technical abilities to successfully implement the project: think also about sustaining communication and cooperation, understanding the data, time management, presentation, identifying and understanding problems, analysing results etc.

Working in groups allows you to pick more challenging tasks and create more interesting projects. Remember that the benefit of group projects comes from actually working together, and learning from each other. Really successful group projects draw on each team member's strengths and skills and aren't just made up of several smaller "parallel" projects.

After formation, your group is required to make a group contract that will govern your collaboration. The contract will specify expectations for the group members as well as procedures for dealing with any issues that may arise. Your contract should answer (at least) the following questions:

- Which task do you want to work on?
- How will you communicate with each other? (Remember that some of you may not be physically present over teaching break.)
- How often and where will you meet?
- How will you make sure that your meetings are productive?
- What will you do if a group member breaks any rules in this contract (e.g., doesn't show up for a meeting)? Take this seriously, and define conditions under which a member can be excluded. Things happen, and if one partner can't fulfil their obligations, neither should they be carried along, nor should it endanger the whole group project.

Please really think about this because you all depend on each other for the success of the project.

Hand in a one page document that addresses these questions and is signed by everyone in the group. (Plus names in block print + student ID numbers.)

3. Planning paper (individual)

The planning paper serves two purposes. On the one hand, it is a short project proposal which specifies what you are going to do, how, and why. On the other hand, it will require each team member to reflect on what you are hoping to learn from the project and what you will contribute to its success. The planning paper is a short (max. 2 pages/1000 words) document, which **each student will write and hand in independently**.

The following questions should be answered in your planning paper:

- What is your project about and what approach are you going to take? (At least, insofar as you can see now; it may of course turn out that you have to adapt this once you have started to work on the project full time.)
- What tools and/or data will you use? (Same here: later adaptations will be allowed, of course.)
- What will you learn from this project? How does the successful implementation of the project help you achieve the learning objectives of the class?
- What will be your personal role in the team, which skills and what knowledge can you contribute?

Note: Please note that we consider this planning paper (as well as all other assignments/papers you hand in) as an individual exam. You must therefore write every word of it yourself. It is unacceptable to copy without attribution anything from a book, paper, or the web, or your project partners' papers! This is considered plagiarism and is subject to be reported. Please describe your project and any methods you use in your own words (demonstrating your own understanding of it).

You must use ACL's template (only the Latex one is allowed!) and citation format available in [Overleaf](#) or on [this repository](#). Do not change the template part in any way.

4. Project presentation (group)

In the last week of class, each group will briefly present their project. These presentations serve a double purpose: First, to get you started well before the final deadline and help you evaluate (while there's still time) whether your idea and approach is going to work. Second, to foster collaboration and let you get to know the others' projects. This way, you will learn not only from your own project but also get to see (preliminary versions of) the cool things the others are doing. **Each presentation should be no longer than 10 minutes**, and you may designate one presenter or split it up among you. Your presentation will likely only include preliminary results (from previous work or a baseline). You should also provide a clear motivation and problem statement, a description of your approach, and your plan on how to evaluate your project. It is typically necessary to refer to previous work, as well, in order to provide context, background, and ideas for what works and doesn't work.

5. Project implementation (group)

The final project should be handed in as documented, running Python code. (There are no restrictions on frameworks that you can use; but of course you need to properly acknowledge any code you take from elsewhere and any frameworks that you import and build on. And of course you are not allowed to use existing solutions to the same task!) You must implement your solution using either logistic regression with multiple well-defined features or neural networks.

You must use [GitUP](#) and give Sherzod (@hakimov) and Brie (@madureiralasota) access to your repository. Please also include a README with instructions on running the code (and thus verifying the results). Proper code and deployment documentation is an important criterion for your submission. Ideally, provide a [Docker](#) file that encapsulates your whole implementation and all requirements so that results can be reproduced with a single call. Include all code you used (including any preprocessing scripts, evaluation scripts, as well as scripts you used to generate tables and plots) and no code that you did not use.

Here are two useful tutorials on building research software that you may find useful: [The Good Research Code Handbook](#) and [Research Software Engineering With Python](#).

6. Project report (individual)

The project report is a paper summarizing your project and reflecting on your individual learning outcomes. It is a coherent essay whose length should be between 5 and 9 pages of content (not less, not more), plus unlimited pages for references and appendix. References do not count as content here. In your paper, please answer the following questions:

- Work report part
 - What was the topic of your group's project, your approach and main results?
 - How did you evaluate the results and what are your conclusions?
- Your contribution (in Appendix)
 - How did your project support you in obtaining your learning objectives (see planning paper)?
 - What did you personally contribute to the project?
 - What are the main take-away messages from your project? What did you learn about your chosen topic?

The project report should stand on its own (i.e., it should not assume familiarity with the task or with your planning paper), but you may reuse material from your previous paper if you want. It is both a report on what you did, as well as a reflection of what you learned in your project.

Make sure to check and follow, whenever applicable, the [ACL guidelines on responsible NLP research](#).

You must use ACL's template (only the Latex one is allowed!) and citation format available in [Overleaf](#) or on [this repository](#). Do not change them in any way.

6.1 Paper Structure

1. **Abstract:** summarizes your work and main conclusions (maximum 300 words).
2. **Introduction:** explains the motivation of your work. The motivation is not that this is a course requirement; we want to see that you understand the importance of this task.
3. **Related Work:** discusses relevant literature (3-4 works) that are relevant or directly related to your approach. How does the proposal build on earlier work? In what sense does it go beyond that? (Often, authors only list papers that are about similar problems. That is not very helpful.)
4. **Task Formalisation:** How is the problem formally framed / turned into a problem that the applied methods can handle?
5. **Data:** describe the dataset you are using, including relevant descriptive statistics and a few samples, and provide a short analysis based on the data statements proposed by [Bender and Friedman \(2018\)](#).
6. **Experiments:** discusses the models, baselines, the implementation and the proposed evaluation procedure. Detailed description of the model architecture tested in the experiment, together with baseline models (earlier work, or some agreed upon "simple but good" approach) for comparison. Goal must be reproducibility; reader should be able to re-implement model based on information in paper.
7. **Results:** presents the results (using nice plots and tables) and the error analysis, including the linguistic aspects.
8. **Conclusion:** summarizes what you did and the main takeaways of your project.
9. **Limitations and Ethical considerations:** discusses the limitations, ethical considerations and other relevant content related to the responsible NLP material we covered during the course. For instance: the linguistic phenomena of the task; the dataset license, collection, use and re-use; the societal impact of the original paper and/or the reproduced results; potential harms, risks or dual use of the findings; aspects of transparency, fairness, explainability, accountability; limitations of the original paper and of your findings; environmental impact; suitability of the evaluation metrics; ethical considerations; reproducibility issues. This section can extend over the 9th page if necessary.
10. **References:** lists all the scientific references you cited in your paper.
11. **Appendix:** any supplementary material (e.g. reproducibility details) not necessary in the main text, meaning that the reader can skip it if they want to, without missing anything relevant. And your

personal takeaways.

Check the generic paper structure we provided you for more details about each section.

6.2 Academic Writing

The papers you submit wrt. your final project are part of a specific genre, academic writing. Student papers are special since they serve the purpose of clearly documenting what you did (as in scientific papers), but also to demonstrate what you know/have learned. Please remember that your instructors can only grade you based on what's written in your papers – so try to make it easy on us to understand what you know and what you have achieved. Initially, it can be hard to judge which things are “obvious” and shouldn't be explained and which things need repeating (even though we both know them). It usually helps to picture an intelligent peer who hasn't taken this class yet as your target audience. I.e., you can assume most basic knowledge about the field (of the kind that undergraduates possess) as given, but should explain or at least mention anything that is beyond that level, though briefly.

In terms of style, academic writing is very results-oriented, focusing on outcomes and argumentation much more than on the process. The most common mistake by students with little university-level writing experience is to faithfully describe what they did (and how they changed their approach in the course of the project) in largely temporal order. Instead, focus on methods and results (it matters much less how you got there, especially when you don't have much space). We recommend stating your main result early on (in the introduction)---“an academic paper is not a crime novel.”

All non-obvious statements need support, either by argumentation (why did you conclude this?) or by citation. Make sure to cite appropriate literature. It is clear that you won't have time for a thorough literature review, but in many cases there are straightforward previous approaches, and if you rely on them, you must cite them correctly. The same applies to code packages and data sets that you use in your project.

Finally, respect your instructors' time by proofreading your papers.

7. Submissions

There will be a link on Moodle for each to-do. Here is what you have to upload by each deadline:

Milestones and Requirements.

Date	To-Do
Jan 09, 2022	Signed group contract due (not graded), one PDF document per group
Feb 06 and 09, 2023	Project (plan) presentations (in person, in groups)
Feb 10, 2023	Planning paper, one PDF document (individual)
Mar 31, 2023	Project report due, one PDF document (individual)

All deadlines are in Berlin time zone.

The report should contain a link to the code repository and the commit hash of the commit that counts. You must use [GitUP](#) and give Sherzod (@hakimov) and Brie (@madureiralasota) access to your repository.

8. Questions

Feel free to ask questions on Moodle.

9. Evaluation

The grading for the project will be based on the written report and on the code+documentation. The grade, in general, can take into account the criteria below (based on Sharid Loáiciga's guidelines). As we cannot exhaustively anticipate all possible problems, other aspects may also emerge during grading.

Formalia

- Are you following all the instructions and guidelines in this document?
- Did you use the right template without any modifications?
- Are the citation style and format correct?
- Is the formatting consistent and does it make the paper easy to read?

Understanding and correctness

- Rationale and goals of the project are introduced: Are there any problems in conceptualizing the problem, and mistakes in your understanding of basic issues?
- Related work is assimilated and tied to the research question: Do you adhere to academic standards by citing relevant papers and comparing to them?
- Methods are understood: Is anything you wrote in your report blatantly wrong? Are the algorithms/methods described correctly?

Quality of the report

- Information is organized coherently: Is there a coherent thread throughout the paper from the motivation through approach to results?
- Ideas are communicated accurately and appropriately: Is the style formal, scientific, precise, with no spelling errors?
- Graphs and tables: Do you include visual supports such as graphs and tables? Do you refer to all of them in the text? Do they support your arguments / display your results? Are they nicely and carefully formatted? Do they have self-contained, descriptive captions?
- Quantitative and qualitative analysis: are the results explained and analysed?
- Do you show a critical approach and a good understanding when assessing other papers and when discussing your results, limitations of your work, ethical considerations, dataset exploration, error and model analysis?

Code

- You provide enough information to recreate the Python environment.
- Code runs with no issues.
- Code is properly and thoroughly documented, organized, easy to read and well-structured, so it can be understood and run or adapted by someone who has read your paper (or yourself in one year).
- Results can be verified in inference mode (please submit all necessary materials to check your system).