Our project step by step:

**Step 0:** Finding datasets – silver data and large gold data

**Angle 1:** Comparing the results of silver data to ground truth

**Step 1:** Re-annotate the silver data to get ground truth

**Step 2:** Compare ground truth to silver data created labels, to see where errors lie, what type they are, and how often they occur.

**Step 3:** Create performance metrics. Where do we disagree? Where does the model disagree with us?

**Angle 2:** The accessibility and usability of silver data

**Step 1:** Divide gold data into training, validation and test sets, strip test set of labels

**Step 2:** Create a NER model (or use existing model), to label the data (without labels)

**Step 3:** Compare results with ground truth gold data labels

**Step 4:** Argue about time spent

Topic: Quality check of silver (or gold) data

* Re-annotate existing data manually, without looking at existing annotations
* Check where the disagreements are between human annotators and check if they overlap with the difficult cases for automatic systems (**analyzing where human annotators disagree** and **checking if those cases are also difficult for NER models)**

Gold data = **Manually annotated by experts**, following strict guidelines. Considered the highest quality.

Silver data = **Automatically labeled (or weakly supervised)** using AI models, heuristics, or crowd workers. May contain errors.

**Steps:**

1. Analyze human annotation disagreements (use a NER dataset with many annotators)

* Different entity labels for the same text
* Measure agreement (Inter-Annotator Agreement) using Cohen’s Kappa, Fleiss’ Kappa, or Krippendorff’s Alpha
* Find patterns

2. Compare with automatic system errors

* Train a NER on the dataset
* Find what it struggle with

3. Find the Overlapping “Hard Cases”

* Check if **human disagreements** overlap with **model errors**
* Identify **linguistic patterns**

Possible datasets:

Gold standard:

Silver standard:

* MultiCoNER II (recommended by Rob in the lecture about project topics)
* WikiAnn (recommended by Rob in the lecture about project topics)

Synthetic dataset methods:

* SpaCy
* Snorkel

Notes from Rob at the presentation:

How do we evaluate the data?

Find large already-annotated data s.t. we have a ground truth to compare to.

Important dates:

* **11/3-2025:** Important lecture, presentation on different possible topics and exam information
* **13/3-2025:** day 1 of groups presenting project proposals
* **20/3-2025:** Deadline for baseline predictions (???)
* **27/3-2025:** day 2 of groups presenting project proposals
* 1/4-2024: important lecture, TA presentation of previous years’ projects
* **3/4-2025:** Written project proposal hand-in date
* **8/4-2025:** Important lecture, how to write a good NLP report (according to Rob)
* **16/5-2025:** 9:00 am, possible early hand-in for preliminary feedback
* **23/5-2025:** final hand-in for the project
* **23/6-2025 – 26/6-2025**: oral exams

Email from Rob 7/3-2025:

Upload slots for your pdfs can be found on LearnIt on the bottom of the page (note that if you want to use slides you have to upload them there so I can merge them together to save a lot of time with connecting laptops) If you want to present without slides, you can e-mail me. The options are 13-03 and 27-03 during the normal lecture hours (10:00-12:00).

Next Tuesday (11-03) I will present potential project topics/directions. However, you do not have to pick from these, you can also propose your own. The information about the project proposal and final assignment will be added to LearnIt by the end of today, together with the slides for Tuesday.

Note that the presentation is mandatory to start your project, the goals are

* To see what the other groups are doing (so please join at least the full hour of your presentation slot). Note that you can collaborate across groups as well, for example to annotate a dataset together.
* To check whether your direction makes sense, so that you do not have to write a full proposal, do literature research etc.
* To get feedback. E.g. perhaps there is a toolkit or a dataset that can help you out, and save you a lot of time.

Note also that the presentation is just 5 minutes per group, so I would suggest to skip table of contents and extensive background overviews. Focus on what you contribute instead (rule of thumb is that most people need 1 minute per slide).

Exam info:

**Time**:

* 10 minutes of group project presentation
* 10-12 minutes individual per person
  + Some questions about the project
  + Focus on 1 random (main) topic from the exam syllabus
  + There are extra topics we turn to if necessary: Tokenization, bias, experimental standards
  + Expect a variety of questions, like:
    - Walk us through algorithm/method X
    - How does method X differ from method Y?
    - What are the benefits of using method X for task Z?
    - How can we best solve Z?
  + What to remember:
    - Main terminology, but more importantly: intuition, reasoning and motivations
    - Often used and basic formulas: RNN, transformers, Kappa, Naive Bayes, Logistic regression, Laplace smoothing, temperature, F1
    - Algorithms and architectures: LSTM, SGD, Viterbi, Min edit distance, ...
    - Skills: Basic POS tagging, regex, ...
    - Strengths, weaknesses and limitations of methods

From the slides (11/3-2025):

**Requirements**:

* Explore something new (could be a reproduction of an existing study, but it has to say something new, and be new for us)
* Hand in should be max. 5 pgs ACL style PDF, see link: <https://github.com/acl-org/acl-style-files>
  + References/bibliography and appendix do not count (thank god!), and Rob says we can add details of setup, extra results etc. As an appendix. Report does need to be readable without the appendix.
* Focus: NER (Named Entity Recognition)
* Topic can be anything, but Rob has given us some inspiration. If we want to choose our own topic instead of one of the suggested, we send Rob an email
* All groups are required to implement a Baseline with EWT data

**Baseline predictions (20/3-2025):**

* Needs to be more advanced than most-frequent class (which is just O). Note that span-f1 is 0.0
* Viterbi is not a good choice, transition probabilities are uninformative
* A better idea might be LSTM or BERT, see assignment solutions

**NER**:

* Identifying class of a word (entity). Assigning that class
* Note that entities can be nested
* Note that labels may differ across datasets
* See 3 examples below:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Barack | Obama | was | born | in | Hawaii |
| B-PER | I-PER | O | O | O | B-LOC |
|  |  |  |  |  |  |
| IT | University | of | Copenhagen |  |  |
| B-ORG | I-ORG | I-ORG | I-ORG |  |  |
| O | O | O | B-LOC |  |  |
|  |  |  |  |  |  |
| Barack | Obama | was | born | in | Hawaii |
| B-POL | I-POL | O | O | O | B-GEO |

* Good resource for NLP research papers for inspiration: <https://aclanthology.org/>
  + Most influential are: NAACL, ACL, EACL, AACL, EMNLP, (findings/LREC)

**Evaluation with span-F1:**

* Precision and recall over spans instead of words
* Conlleval.pl is the most common implementation
* Micro-F1 considers instances instead of labels
* Also reports per class scores
* Rob also provides an implementation: span\_f1.py, known to give different results. Also returns:
  + Loose F1
  + Unlabelled F1
  + Can be used to analyse whether a model is bad at finding the exact boundaries or the labels.

**Datasets: all except wikiann and multiconer II can be found in the repository**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Langs** | **Domains** | **Nested?** | **Human?** | **# labels** |
| EWT | EN | 5 web | Y | Y | 12 |
| Tweebank | EN | Social | N | Y | 4 |
| DaN+ | DA | News, social | Y | Y | 12 |
| CrossNER | EN | 5 wiki | N | Y | 39 |
| Universal NER | 17 | Mixed | N | Y | 3 |
| WikiAnn | 295 | Wiki | N | N | 3 |
| MultiCoNER II | 12 | Wiki | N | N | 33 |
| Few-NERD | EN | Wiki | N | Y | 66 |

* WikiAnn and MultiCoNER II are silver data, and we are warned to use with caution.
* If insufficient, find another or make one. How to get:
  + Search for links in paper/google/author’s website
  + Search on huggingface
  + Ask Rob
  + Email authors, put Rob in CC

**Project ideas if we want to pick one of the existing topics from Rob:**

General direction: can we improve NER systems? Might be hard though, so instead find challenging situations to focus on. Maybe incorporate POS tags?

* Checklist evaluation
* Predict cross-X performance
* Multi-dataset NER
* NER for new language/domain (cross-dataset)
* Quality check of silver (or gold) data
* Emerging NEs
* Active learning for spans
* BILSTM vs Discriminative vs Generative LMs
* Slot and intent detection
* Is supervised NER solved?
* Few-shot NER
* Unsupervised NER
* ChatGPT (not recommended)

**Metrics for evaluating the quality of annotations**

* Cohen’s Kappa (code- week 2 solution)
  + Statistical measure of inter-annotator agreement for categorical data
  + adjusts for the agreement that could happen by chance
  + by breaking down per label, its possible to find which labels are causing the most trouble
  + reflects how balanced classes are (might get high accuracy because annotators agree on the 0 label, but still get low kappa)
  + We need to change them back into labels (?)
* Exact Match Accuracy: Percentage of entities where all attributes (span and label) match exactly.

Once you've defined how to count matches, you can compute:

* Precision = correct predictions / total predictions
* Recall = correct predictions / total gold annotations
* F1-score = harmonic mean of precision and recall
  + We should use it since there is not equal class distribution (most labels are 0)
* You can calculate:
* Micro-averaged: Treats every prediction equally (good for imbalanced labels)
* Macro-averaged: Averages per-label scores (helps you see which labels are harder)
  + compute the performance for each class, and then average over classes
  + more important when performance on all the classes is equally important