EDAN70
Project in Computer Science
https://cs.lth.se/edan70/
Project in Artificial Intelligence

Pierre Nugues
Pierre.Nugues@cs.lth.se

March 24, 2025

Projects

1. Define a study topic and an application in natural language processing. You may define it yourself or with the help of the instructor.
2. Do a quick survey the relevant literature. Start from one or two papers, and, if appropriate, existing code
3. Select datasets, algorithms, and outline an implementation strategy and timeline
4. Iterate:
   - Implement a prototype. This should be started the first week.
   - Evaluate it
5. Write a project report. This should be started the first week and shared through `https://www.overleaf.com/` (with Latex and Bibtex) using a conference template (ACL 2025)
6. Release your code (optional)
7. Submit paper to a conference (optional). No funding for conference fees and travel.

Organization

- The project will take place in the 4th LP.
- There is no dedicated location for it.
- The participants will work on the machines in the basement or on their own machines.
- Experimentally, I have opened a NAISS account (no guarantee)
- The duration of time spent on the project should be of about two weeks.
- Each participant should collaborate with one possibly two other people.
- Weekly progress meeting with your instructor. We will show your prototype as early as the second week.

As a result of a work accident, I have a strong tinnitus that is very annoying. I will not take more than five groups.

Possible Subjects

1. Projects building on the assignments
2. Projects connected to research
3. Exploratory project on typography.
4. Finally, projects from yourself

Assignments

1. Classification
2. Subword segmentation
3. Language detection and text categorization
4. Sentence embeddings (SBERT)
5. Translation

Example Building on an Assignment

Translation with transformers:

1. Larger corpus, subwords with sentencepiece, etc.
2. Complete implementation: learning rate, smoothing, etc.
3. Maybe introduce some variants like RMS normalization and other details
4. Try with a decoder-only stack
5. Train with more computing resources (if possible)

Research

Entity linking:

1. Associate words or definitions with things from the real world
2. Use Wikidata as entity repository
3. Interested in (historic) encyclopedic sources
4. Interested in knowledge construction and evolution

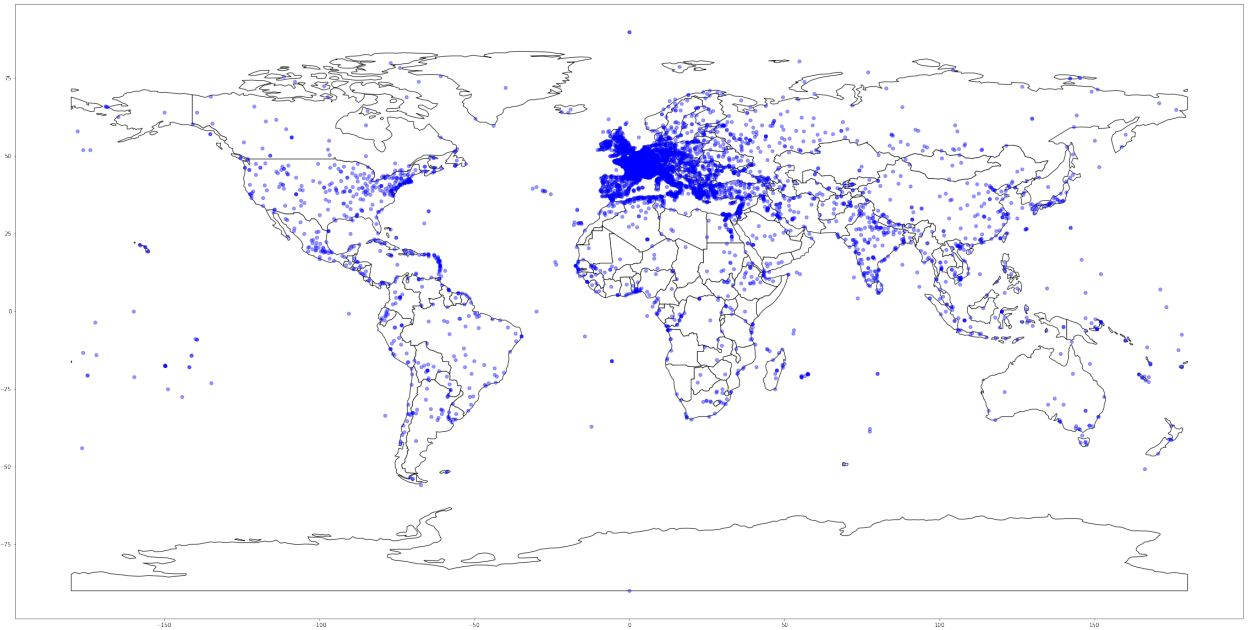
Examples

From a historic French dictionary:

> **Lund**, *v. de la Suède méridionale, sur le Hœje ; 17.000 h. Université célèbre.*

`http://nenufar.huma-num.fr/?article=81833`

Wikidata: `https://www.wikidata.org/wiki/Q2167`



Examples

Form historic Swedish encyclopedias:

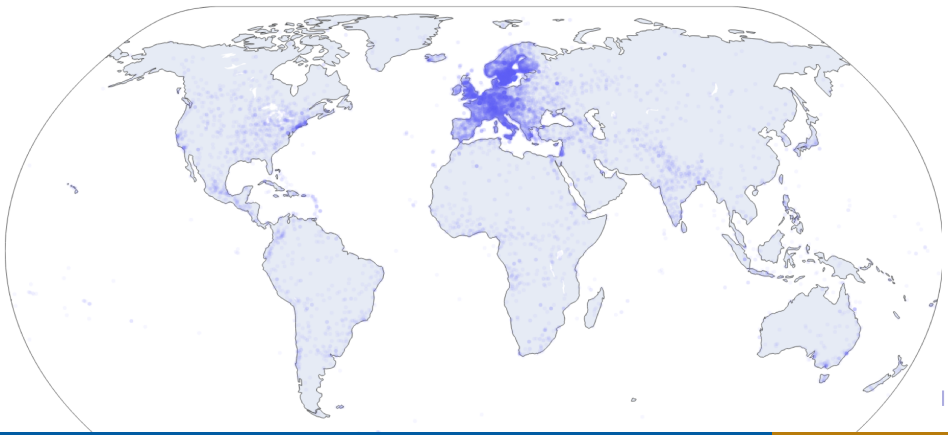
> **Lund**, *uppstad i Malmöhus län med rikets andra universitet och säte för biskopen öfver Lunds stift, är beläget omkr. 8 km. från Öresund...*

`https://nordiskfamiljebok.dh.gu.se/article/1/45575`

> **Lund**, *uppstad i Malmöhus län med rikets andra universitet och säte för biskopen öfver Lunds stift, är beläget vid 55° 41' 52" n. br. och 4° 52' 15"...*

`https://nordiskfamiljebok.dh.gu.se/article/2/183416`

`https://aclanthology.org/2024.lrec-main.962/`



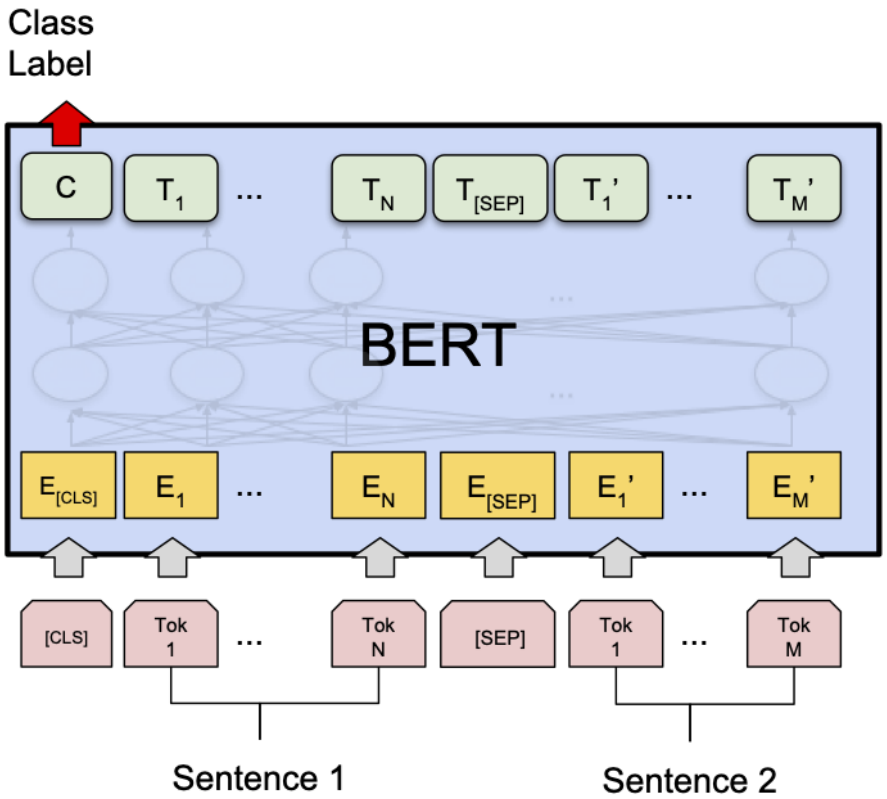
Project Proposals (I)

1. Categorize the entities from their definition: person, location, etc.
2. Track and match entities across editions of dictionaries: https://runeberg.org/nf/
3. Link entities extracted from text.

Methods:

- Using existing tools (SpaCy)
- Using BERT: https://github.com/lajanugen/zeshel or ReFinED (https://arxiv.org/pdf/2207.04106)
- Using vector databases
- Deduplication with hashcodes: https://aclanthology.org/2020.lrec-1.494.pdf

Application: Sentence Pair Classification



(a) Sentence Pair Classification Tasks: MNLI, QQP, QNLI, STS-B, MRPC, RTE, SWAG

from Devlin et al., *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*, 2019

Vector Databases

1. Represent definitions with sentence embeddings (SBERT)
2. Measure the similarities

Tools:

- SBERT in Swedish: https://kb-labb.github.io/posts/2021-08-23-a-swedish-sentence-transformer/
- SBERT benchmarks: https://huggingface.co/spaces/mteb/leaderboard
- Emergence of vector databases: https://www.pinecone.io/, https://milvus.io/, https://qdrant.tech/, https://www.trychroma.com/, etc.

Project Proposals (II): Classification

1. Corpus of medical reports
2. Predict if a heart transplanted patient will survive more than one year (or five years)
3. https://ieeexplore.ieee.org/document/9871788
4. Collaboration with Johan Nilsson

Project Proposals (III): Baby Language Models

1. Large language models are unsustainable for most of us: Meta uses 600,000 H100 (https://engineering.fb.com/2024/03/12/data-center-engineering/building-metas-genai-infrastructure/)
2. BabyLM challenge: 10M or 100M words https://babylm.github.io/
3. Needs high-quality texts: Example of Chi: *Textbooks are all you need* https://arxiv.org/abs/2306.11644
4. Simple modifications can have a big impact: https://aclanthology.org/2023.conll-babylm.20/

Exploring BabyLM

1. Adapt a BabyLM architecture and train it on an encyclopedia
2. Train on one edition of *Nordisk familjbok* and see the results on an other edition
3. See how well it performs on downstream tasks
   - Train on a medical encyclopedia
   - Evaluate on the classification of diseases
4. The BabyLM from Charpentier is an encoder (BERT). See if it could be adapted to a decoder

Project Proposals (IV): Document Structure

1. Historic encyclopedia are digitized using OCR
2. Last year, we extracted the entries using ad-hoc rules
3. Explore automatic ways to recover the document structure:
4. Examples to start with:
   - DocBank https://arxiv.org/pdf/2006.01038
   - LayoutReader https://arxiv.org/abs/2108.11591
   - https://openaccess.thecvf.com/content/WACV2022/papers/Wang_Post-OCR_Paragraph_Recognition_by_Graph_Convolutional_Networks_WACV_2022_paper.pdf

Others

1. If somebody is interested in typography, tell me.
   - https://openaccess.thecvf.com/content/CVPR2023/papers/Wang_DeepVecFont-v2_Exploiting_Transformers_To_Synthesize_Vector_Fonts_With_Higher_Quality_CVPR_2023_paper.pdf
   - https://openaccess.thecvf.com/content/CVPR2023/papers/Xia_VecFontSDF_Learning_To_Reconstruct_and_Synthesize_High-Quality_Vector_Fonts_via_CVPR_2023_paper.pdf
2. Finally you can propose your project:
   1. ?
   2. ?