IN5550: Neural Methods in Natural Language Processing
Introduction

Andrey Kutuzov, Vladislav Mikhailov, Roxana Pop,
David Samuel, Sondre Wold & You

University of Oslo

January 20, 2025



Today

- Introduction
- Why a course on neural methods for NLP?
  - Motivation
  - Historical trends
  - Some contrasts between NNs and traditional ML
  - Success stories
- Course overview
  - Course content
  - Programming environment
  - Lab sessions and obligatory assignments
  - Home exam

Who we are?



Andrey Kutuzov, Language Technology Group



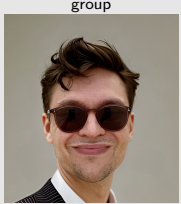
Sondre Wold, Language Technology Group



David Samuel, Language Technology Group

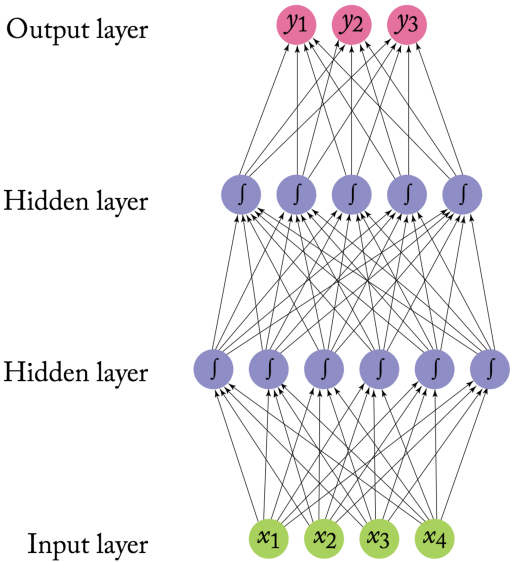


Roxana Pop, Analytical Solutions and Reasoning group



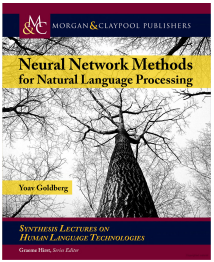
Vladislav Mikhailov, Language Technology Group

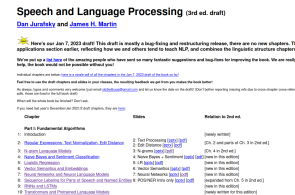
What is a neural network (NN) model?



- Neural networks (NNs) are a family of powerful machine learning models.
- Weakly based on the metaphor of a neuron.
- Non-linear transformations of the input in the 'hidden layers'.
- Learns not only to make predictions, but how to represent the data
- 'Deep Learning': NNs with several hidden layers.

Knowledge sources




Speech and Language Processing (3rd ed. draft)

Dan Jurafsky and James H. Martin

Here's our Jan 7, 2023 draft! This draft is mostly a bug-fixing and restructuring release, there are no new chapters. The restructuring moves the applications section earlier, reflecting how we and others tend to teach NLP, and combines the linguistic structure chapters in one section.

We've put up a list here of the amazing people who have sent us many fantastic suggestions and bug-fixes for improving the book. We are really grateful to all of you for your help, the book would not be possible without you!

Individual chapters are below (<u>here is a single pdf of all the chapters</u> in the Jan 7, 2023 draft of the book so far)

Feel free to use the draft chapters and slides in your classes, the resulting textbook we put it into you makes the book better!

As always, typos and comments are very welcome (just email slp3ed-pubs@gmail.com and let us know the date on the draft)! (Don't bother reporting missing refs due to cross-chapter cross-reference problems in the individual chapter pdfs, those are fixed in the full book draft)

When will the whole book be finished? Don't ask.

If you read last year's December/Jan 2021-3 draft chapters, they are here

Chapter	Slides	Relation to 2nd ed.
Part I: Fundamental Algorithms		
1 Introduction		[newly written]
2 Regular Expressions, Text Normalization, Edit Distance	2. Text Processing (ppt) (pdf) 2. Edit Distance (ppt) (pdf)	[Ch. 2 and parts of Ch. 3 in 2nd ed.]
3 N-grams Language Models	3. N-grams (ppt) (pdf)	[Ch. 4 in 2nd ed.]
4 Naive Bayes and Sentiment Classification	4. Naive Bayes + Sentiment (ppt) (pdf)	[new in this edition]
5 Logistic Regression	5. LR (ppt) (pdf)	[new in this edition]
6 Vector Semantics and Embeddings	6. Vector Semantics (ppt) (pdf)	[new in this edition]
7 Neural Networks and Neural Language Models	7. Neural Networks (ppt) (pdf)	[new in this edition]
8 Sequence Labeling for Parts of Speech and Named Entities	8. POS/NER Intro (ppt) (pdf)	[expanded from Ch. 5 in 2nd ed.]
9 RNNs and LSTMs		[new in this edition]
10 Transformers and Pretrained Language Models		[newly written for this edition]
11 Fine-tuning and Masked Language Models		[new in this edition]
12 Prompting and Instruct Tuning		

- The field is developing rapidly: no definitive textbook exists
- *'Neural Network Methods for Natural Language Processing'* [Goldberg, 2017]
  - For more foundational topics
  - Free e-version available through UiO; `http://oria.no/`
- *'Speech and Language Processing'* by Dan Jurafsky and James Martin (online, 2025)
  - For more advanced and recent topics
  - `https://web.stanford.edu/~jurafsky/slp3/`
- Supplementary research papers will be added.
- Check the schedule for the syllabus for each lecture!

Paradigm shifts in NLP (and AI at large)

- 50s–80s: mostly rule-based (symbolic / rationalist) approaches.
- Hand-crafted formal rules and manually encoded knowledge.
- (Though some AI research on neural networks in the 40s and 50s).

Paradigm shifts in NLP (and AI at large)

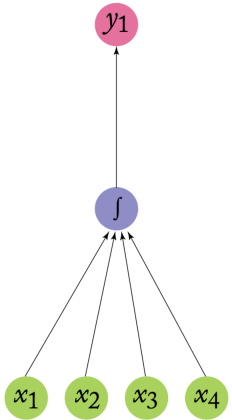
- 50s–80s: mostly rule-based (symbolic / rationalist) approaches.
- Hand-crafted formal rules and manually encoded knowledge.
- (Though some AI research on neural networks in the 40s and 50s).
- Late 80s: success with statistical ('empirical') methods in the fields of speech recognition and machine translation.
- Late 90s: massive shift towards machine learning (ML).
  - Based on automatically inferring statistical patterns from data.
- 00s: Machine learning methods dominant.

Paradigm shifts in NLP (and AI at large)

- 50s–80s: mostly rule-based (symbolic / rationalist) approaches.
- Hand-crafted formal rules and manually encoded knowledge.
- (Though some AI research on neural networks in the 40s and 50s).
- Late 80s: success with statistical ('empirical') methods in the fields of speech recognition and machine translation.
- Late 90s: massive shift towards machine learning (ML).
  - Based on automatically inferring statistical patterns from data.
- 00s: Machine learning methods dominant.
- 2010–present: neural methods increasingly replacing traditional ML.
- A revival of techniques first considered in the 40s and 50s,
- but recent developments in computational power and availability of data have given great breakthroughs in scalability and accuracy.

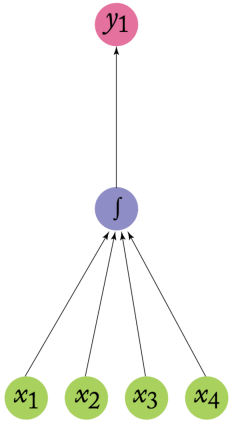
What is a machine learning classifier?

- Very high-level:
- A learned mapping from inputs to outputs.
- Learned from labeled examples; a set of objects with correct class labels.
- 1st step in creating a classifier; defining a representation of the input!
- Typically given as a feature vector.



What is a machine learning classifier?

- Very high-level:
- A learned mapping from inputs to outputs.
- Learned from labeled examples; a set of objects with correct class labels.
- 1st step in creating a classifier; defining a representation of the input!
- Typically given as a feature vector.



'*A good representation does most of the work.*'

[Anderson and Rosenfeld, 1988]

Traditional ML; based on feature engineering

- The art of designing features for representing objects to a classifier.
- Typically also involves large-scale empirical tuning to identify the best-performing configuration.
- Although there is much overlap in the types of features used across tasks, performance is highly dependent on the specific task and dataset.
- Results in very high-dimensional and sparse feature vectors.
- Challenges; discreteness and lack of information sharing.

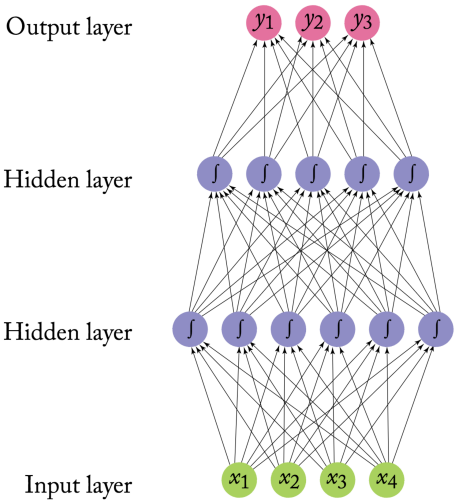
Representation learning

- One of the greatest strengths of neural network (NN) models is their ability to learn their own internal task-specific representations of the data.

Representation learning

- One of the greatest strengths of neural network (NN) models is their ability to learn their own internal task-specific representations of the data.
- One important example within NLP: word embeddings
  - Low-dimensional and dense vector representations of words.
  - Often pre-trained on unlabeled data for a text prediction task;
  - so-called language modeling;
  - the learned word representations can then be used to form the input representations for other down-stream supervised NN models.

Some pros and cons of NNs

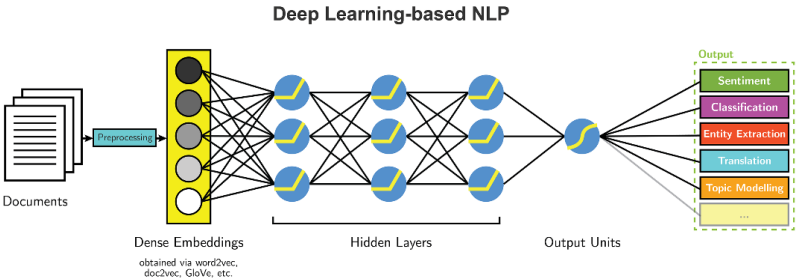


NNs hold promise to...

- reduce manual feature engineering,
- make better use of unlabeled data,
- through the use of distributed continuous input representations,
- and further learn more task-adapted internal representations of the data.
- Tends to scale better with more data.
- Though at the cost of being less interpretable and more complex with more parameters.

Neural methods in NLP

- Most NLP tasks approached as classification problems, using supervised machine learning
  - ...but recently also as text generation.
- Since at least 2014, neural networks completely dominating.
- Very successful on standard NLP benchmarks.



Neural NLP tasks

- Text classification:
  - Sentiment analysis;
  - Spam detection;
  - Language identification.

Neural NLP tasks

- Text classification:
  - Sentiment analysis;
  - Spam detection;
  - Language identification.
- Sequence labeling:
  - Named entity recognition;
  - PoS-tagging.

Neural NLP tasks

- Text classification:
  - Sentiment analysis;
  - Spam detection;
  - Language identification.
- Sequence labeling:
  - Named entity recognition;
  - PoS-tagging.
- Natural language generation:
  - Summarization;
  - Paraphrase generation;
  - Headline generation;
  - Machine translation.

Neural NLP tasks

Machine translation (Google Translate) – neural or statistical?

- No 1:
  Kilimanjaro is a snow-capped mountain with an elevation of 19,710 feet and is said to be the highest mountain in Africa. Its western summit is called "Nga Jengai" (House of God) by the Maasai. Near the western summit lies the desiccated and frozen carcass of a leopard. No one has explained what the leopard was looking for at that altitude.
- No 2:
  Kilimanjaro is 19,710 feet of the mountain covered with snow, and it is said that the highest mountain in Africa. Top of the west, "Ngaje Ngai" in the Maasai language, has been referred to as the house of God. The top close to the west, there is a dry, frozen carcass of a leopard. Whether the leopard had what the demand at that altitude, there is no that nobody explained.

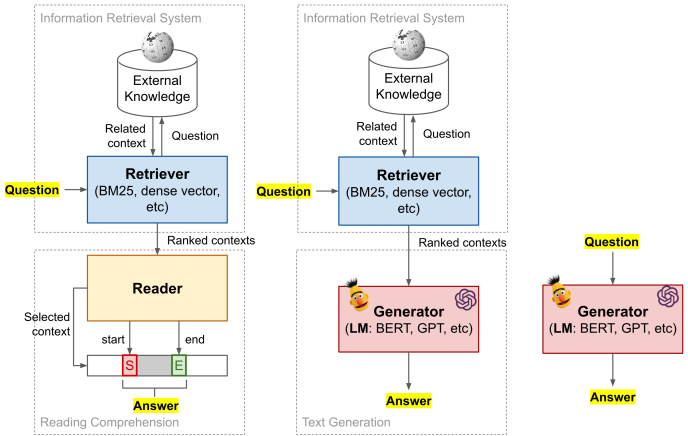
Neural NLP tasks

Machine translation (Google Translate)

- Neural model:
  Kilimanjaro is a snow-capped mountain with an elevation of 19,710 feet and is said to be the highest mountain in Africa. Its western summit is called "Nga Jengai" (House of God) by the Maasai. Near the western summit lies the desiccated and frozen carcass of a leopard. No one has explained what the leopard was looking for at that altitude.
- Ernest Hemingway:
  Kilimanjaro is a snow-covered mountain 19,710 feet high, and is said to be the highest mountain in Africa. Its western summit is called the Masai "Ngaje Ngai," the House of God. Close to the western summit there is the dried and frozen carcass of a leopard. No one has explained what the leopard was seeking at that altitude.

Neural NLP tasks

Question answering



(https://lilianweng.github.io/posts/2020-10-29-odqa/)

Neural NLP tasks

Visual question answering

Who is wearing glasses?

man



woman

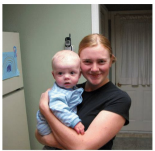


Where is the child sitting?

fridge

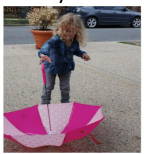


arms

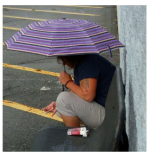


Is the umbrella upside down?

yes



no

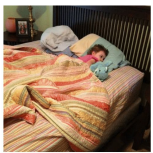


How many children are in the bed?

2



1



(http://visualqa.org/)

Image captioning



"man in black shirt is playing guitar."



"construction worker in orange safety vest is working on road."



"two young girls are playing with lego toy."

(https://towardsdatascience.com/image-captioning-in-deep-learning-9cd23fb4d8d2)

Neural NLP tasks

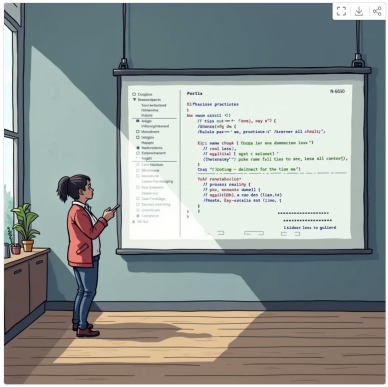
Text-to-image generation

FLUX.1 [dev]

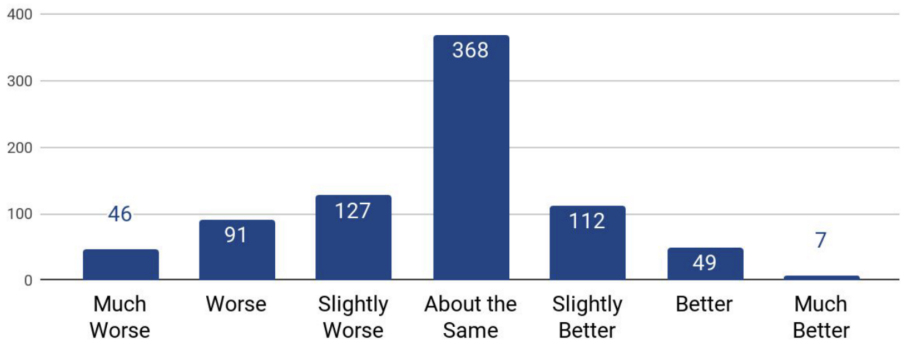
12B param rectified flow transformer guidance-distilled from FLUX.1 [pro]
[non-commercial license] [blog] [model]

Create an illustration of the IN5550 (Neural Methods

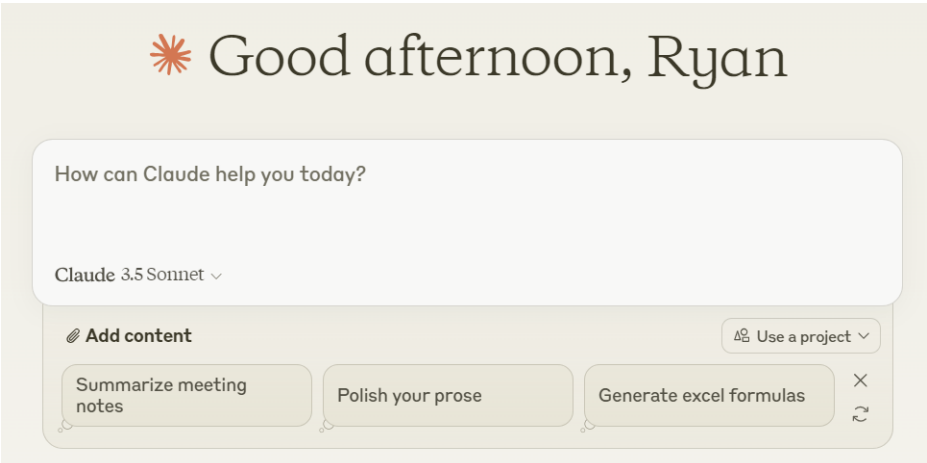
Run



Text-to-Speech [Shen et al., 2018]:



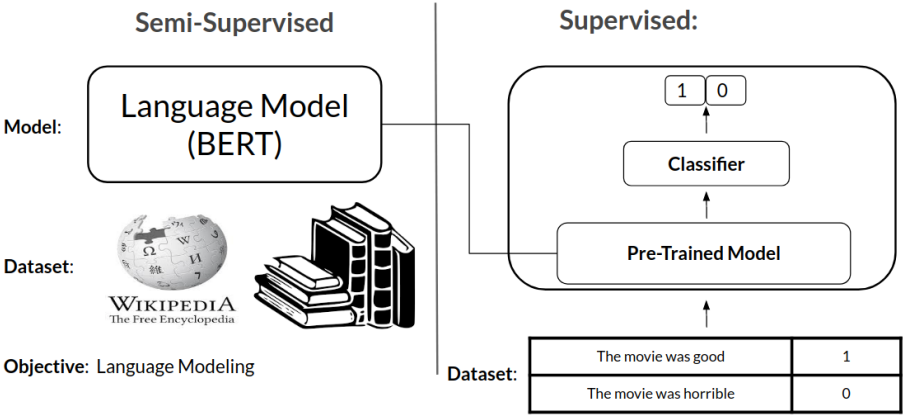
Dialog systems:



Neural NLP tasks

- Neural models have caused great advances in the field of natural language processing, image processing and signal processing
- All of these have converged to the same neural architecture and learning methods
- We will focus only on text $\rightarrow$ but you can easily use the knowledge in other domains

Typical neural NLP system (finetuning)



(https://blog.insightdatascience.com)

Masked Language models

- Given raw (i.e. unlabeled) text as input,
- Masked language Models (MLMs) are trained to predict masked out parts of a sentence

A quick brown ??? jumps over the ??? dog

- 'Self-supervised'

Masked Language models

- Given raw (i.e. unlabeled) text as input,
- Masked language Models (MLMs) are trained to predict masked out parts of a sentence

A quick brown ??? jumps over the ??? dog

- 'Self-supervised'
- Neural MLMs are used as representation learners;

Typical neural NLP system (prompting)

ENG: She is making a mountain out of a molehill.
NOB: Hun gjør en mygg til en elefant.

ENG: Do you remember that baffling murder case?
NOB: Husker du den utrolige mordsaken?

ENG: What's the best way to get there?
NOB: Hva er beste måte å komme seg dit på?

ENG: They say that Tom won't do that.
NOB:

Typical neural NLP system (prompting)

ENG: She is making a mountain out of a molehill.
NOB: Hun gjør en mygg til en elefant.

ENG: Do you remember that baffling murder case?
NOB: Husker du den utrolige mordsaken?

ENG: What's the best way to get there?
NOB: Hva er beste måte å komme seg dit på?

ENG: They say that Tom won't do that.
NOB: De sier at Tom ikke kommer til å gjøre det.

Causal Language models

- Again, given only raw text as input,
- Causal language Models (CLMs) are trained to predict the next word

A

Causal Language models

- Again, given only raw text as input,
- Causal language Models (CLMs) are trained to predict the next word

A quick

Causal Language models

- Again, given only raw text as input,
- Causal language Models (CLMs) are trained to predict the next word

A quick brown

Causal Language models

- Again, given only raw text as input,
- Causal language Models (CLMs) are trained to predict the next word

A quick brown fox

Causal Language models

- Again, given only raw text as input,
- Causal language Models (CLMs) are trained to predict the next word

A quick brown fox jumps

Causal Language models

- Again, given only raw text as input,
- Causal language Models (CLMs) are trained to predict the next word

A quick brown fox jumps over the lazy dog

- These models are not great representation learners, but their ability to generate text gives them a big flexibility

Causal Language models

- Again, given only raw text as input,
- Causal language Models (CLMs) are trained to predict the next word

A quick brown fox jumps over the lazy dog

- These models are not great representation learners, but their ability to generate text gives them a big flexibility
- We can combine finetuning with prompting to teach CLMs to follow instructions

Language models

- A number of different models have been proposed: GPT, BERT, XLNet, T5, PaLM, ChatGPT etc.
- Extreme media attention:



(https://www.nrk.no/kultur/1.16227113)

- Limitations: issues with explainability, sustainability and bias – more about this later in the course

Language models

Our group has been training large language models for more than 5 years



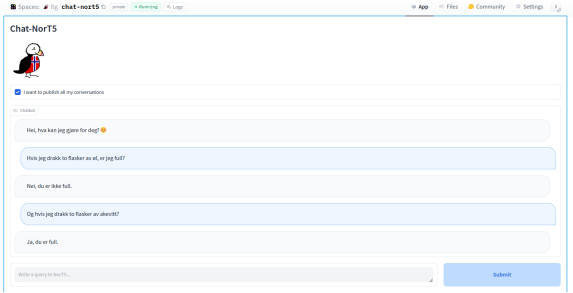
`https://twitter.com/ltgoslo`

Language models

Our group has been training large language models for more than 5 years



https://twitter.com/ltgoslo



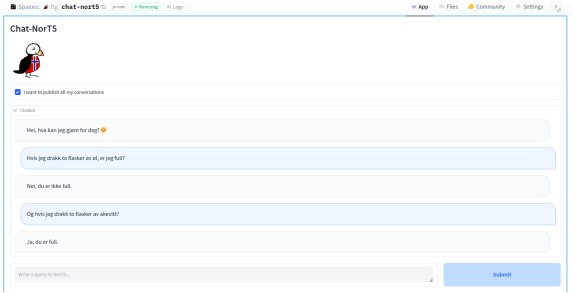
https://huggingface.co/spaces/ltg/chat-nort5, 2023

Language models

Our group has been training large language models for more than 5 years



https://twitter.com/ltgoslo



https://huggingface.co/spaces/ltg/chat-nort5, 2023

- Now, we have a collection of Norwegian LMs of different flavors and sizes
- ...and pre-trained LMs and Transformers lectures are coming later in the course

Getting to Know Each Other

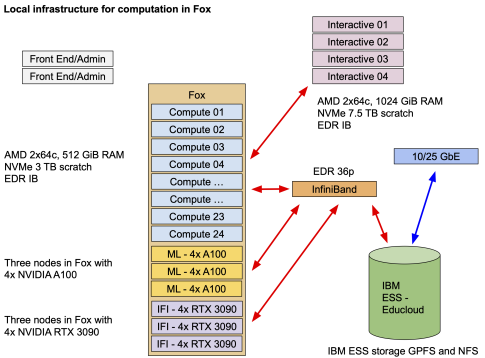
We need to know who you are!

A brief anonymous survey:

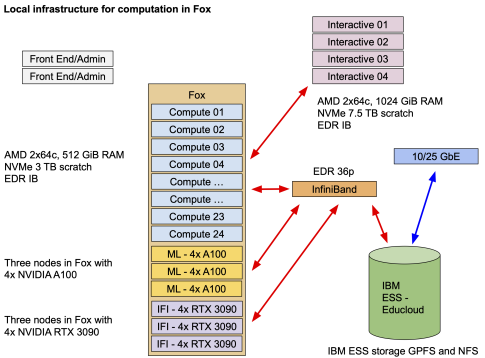
`https://nettskjema.no/a/480940`

Course Content

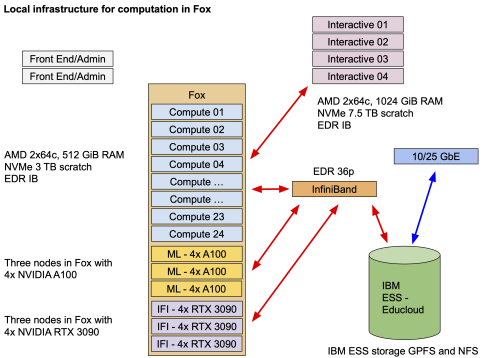
- Training deep neural networks;
- Dense feature representations and word embeddings;
- Language modeling, contextualized embeddings;
- Attention mechanism, Transformer;
- Pretraining modern Transformer-based language models;
- Generation, finetuning and prompting;
- Trustworthiness and Green AI;
- Retrieval-Augmented Generation (guest lecture)
- Mechanistic interpretability.



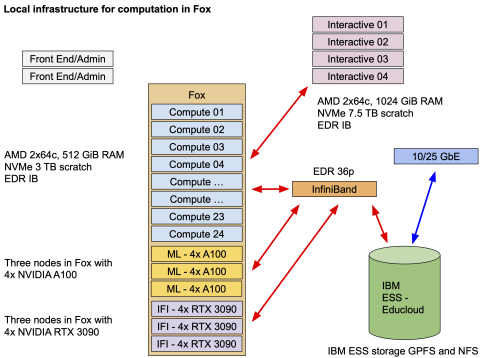
- Strong practical and programming elements in this course;



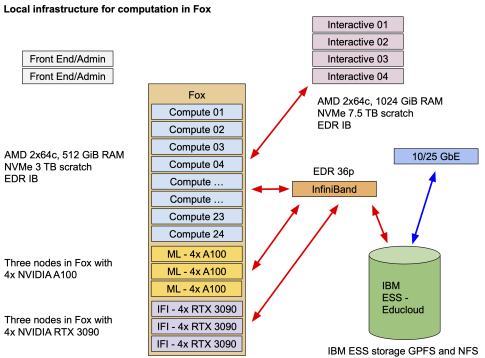
- Strong practical and programming elements in this course;
- three obligatory assignments; all predominantly ‘hands-on’;



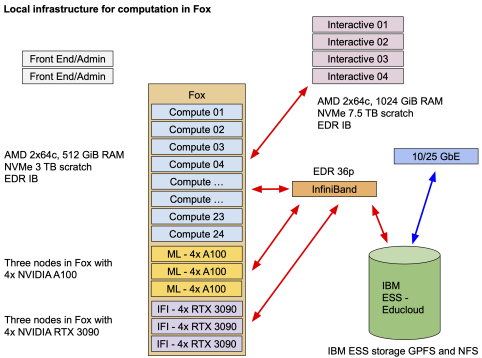
- Strong practical and programming elements in this course;
- three obligatory assignments; all predominantly ‘hands-on’;
- representative of sub-problems in typical MSc theses at the Language Technology Group (LTG);



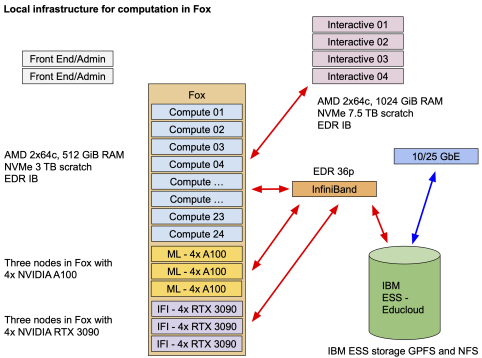
- Strong practical and programming elements in this course;
- three obligatory assignments; all predominantly ‘hands-on’;
- representative of sub-problems in typical MSc theses at the Language Technology Group (LTG);
- fairly data- and compute-intensive throughout the semester;



- Strong practical and programming elements in this course;
- three obligatory assignments; all predominantly ‘hands-on’;
- representative of sub-problems in typical MSc theses at the Language Technology Group (LTG);
- fairly data- and compute-intensive throughout the semester;
- learn how to use the UiO Fox computing cluster



- Strong practical and programming elements in this course;
- three obligatory assignments; all predominantly ‘hands-on’;
- representative of sub-problems in typical MSc theses at the Language Technology Group (LTG);
- fairly data- and compute-intensive throughout the semester;
- learn how to use the UiO Fox computing cluster (3,000$^+$ CPUs and GPUs);



- Strong practical and programming elements in this course;
- three obligatory assignments; all predominantly 'hands-on';
- representative of sub-problems in typical MSc theses at the Language Technology Group (LTG);
- fairly data- and compute-intensive throughout the semester;
- learn how to use the UiO Fox computing cluster (3,000$^+$ CPUs and GPUs);
- practical skills: Linux, batch jobs, experimentation, tuning, ...



- Python is a relatively simple programming language with great popularity for neural network–based machine learning;



- Python is a relatively simple programming language with great popularity for neural network–based machine learning;
- it provides a very convenient, high-level coding with a gentle learning curve; works easily across different platforms;



- Python is a relatively simple programming language with great popularity for neural network–based machine learning;
- it provides a very convenient, high-level coding with a gentle learning curve; works easily across different platforms;
- comprehensive standard library; ecosystem of community-maintained add-on modules with specialized (and optimized) functionality;



- Python is a relatively simple programming language with great popularity for neural network–based machine learning;
- it provides a very convenient, high-level coding with a gentle learning curve; works easily across different platforms;
- comprehensive standard library; ecosystem of community-maintained add-on modules with specialized (and optimized) functionality;
- pretty much everything open-source; we provide ready-to-use ‘virtual lab’ on Fox; in principle possible to install ‘at home’ (for development).

A Complex of Interoperable Modules







- NumPy for efficient multi-dimensional arrays (a.k.a. ‘tensors’) and general linear algebra;

A Complex of Interoperable Modules







- NumPy for efficient multi-dimensional arrays (a.k.a. 'tensors') and general linear algebra;
- PyTorch (by Meta) is a mature software infrastructure for deep learning (built natively in C^{++});
  - Any self-respecting technology giant today develops their own DL framework (Google has TensorFlow, NVIDIA has Megatron, etc);
  - plus a few from university environments;
  - open source $\rightarrow$ community involvement;

A Complex of Interoperable Modules







- NumPy for efficient multi-dimensional arrays (a.k.a. ‘tensors’) and general linear algebra;
- PyTorch (by Meta) is a mature software infrastructure for deep learning (built natively in C^{++});
  - Any self-respecting technology giant today develops their own DL framework (Google has TensorFlow, NVIDIA has Megatron, etc);
  - plus a few from university environments;
  - open source $\rightarrow$ community involvement;
- HuggingFace Transformers for easy handling of ‘pre-trained models’;

A Complex of Interoperable Modules







- NumPy for efficient multi-dimensional arrays (a.k.a. 'tensors') and general linear algebra;
- PyTorch (by Meta) is a mature software infrastructure for deep learning (built natively in C^{++});
  - Any self-respecting technology giant today develops their own DL framework (Google has TensorFlow, NVIDIA has Megatron, etc);
  - plus a few from university environments;
  - open source $\rightarrow$ community involvement;
- HuggingFace Transformers for easy handling of 'pre-trained models';
- all integrated in standard modules on Fox (see the course page).

Lectures

No traditional lectures (except this one)

- Pre-recorded short videos;
- Published: every Thursday afternoon (starting this week);
- watch whenever convenient;
- find them in the course schedule under ‘Schedule’;
- every Monday at 14:15: interactive Q&A in Zoom.



Labs and Obligatory Assignments (*grupper & obliger*)

- Labs: Tuesday, 10:15–12:00 (starting next week);
- three obligatory assignments;
- rigid schedule;
- minimum 1p for each submission;
- minimum 56% of points across all three required to qualify for exam (63% for PhD students), i.e. 18 and 20 points respectively;
- no re-submissions.



Labs and Obligatory Assignments (*grupper & obliger*)

- Labs: Tuesday, 10:15–12:00 (starting next week);
- three obligatory assignments;
- rigid schedule;
- minimum 1p for each submission;
- minimum 56% of points across all three required to qualify for exam (63% for PhD students), i.e. 18 and 20 points respectively;
- no re-submissions.



Mechanics of the obligatory assignments

- Schedule: `https://www.uio.no/studier/emner/matnat/ifi/IN5550/v25/obligatory-assignments/`
- Group work warmly encouraged (1–3 students); please give it a shot!
- Final exam as 'baby' research project in May; again, team work possible (for MSc students).

Home Exam

General Idea

- Our guiding metaphor: Preparing a scientific paper for publication.

Home Exam

General Idea

- Our guiding metaphor: Preparing a scientific paper for publication.

7th IN5550 Workshop on Neural NLP (WNNLP 2025)

Home Exam

General Idea

- Our guiding metaphor: Preparing a scientific paper for publication.

7th IN5550 Workshop on Neural NLP (WNNLP 2025)

Standard Process

(1) Experimentation

(2) Analysis

(3) Paper Submission

(4) Reviewing

(5) Camera-Ready Manuscript

(6) Presentation

For Example: The Annual ACL Conference

Main Conference Papers

Submission template available	~~November 1, 2022~~
Anonymity period for ARR papers	~~November 15, 2022~~
Submission deadline for papers submitted to ARR	~~December 15, 2022~~
Anonymity period for papers submitted through Softconf START	~~December 20, 2022~~
Abstract deadline for Softconf START direct submissions	January 13, 2023
Direct paper submission deadline	January 20, 2023
Anonymity period for ARR papers to be committed to ACL 2023	Februrary 17, 2023
Commitment deadline for ARR papers	March 17, 2023
Author response period	March 17-24, 2023
Notification of acceptance	May 1, 2023
Withdrawal deadline	May 8, 2023
Camera-ready papers due	May 22, 2023

:

Same here: WNNLP 2025 (IN5550 Final Exam)

Three weeks for submissions; two weeks for reviewing and revision.

`https://www.uio.no/studier/emner/matnat/ifi/IN5550/v25/final-exam/`

Home Exam

Our timeline

- Track presentations: April 28th
- Home Exam: April 29th – May 20th
- Paper submission – May 20th
- Review period: May 21st – May 23rd
- Camera Ready: May 30th
- Workshop with oral presentations: June 2nd

Proceedings of the Sixth IN5550 Workshop on Neural Natural Language Processing (WNNLP 2024)

Andrey Kutuzov, David Samuel, Vladislav Mikhailov, Roxana Pop and Sondre Wold (editors)

June 03, 2024

University of Oslo, Norway

Course Communication

- Questions?

Course Communication

- Questions?
  - On-line chat room: https://mattermost.uio.no/ifi-undervisning/channels/in5550
  - GitHub issues: https://github.uio.no/in5550/2025/issues
  - email: in5550-help@ifi.uio.no reaches all course staff.
- submissions, solutions, etc. on the GitHub instance at UiO.

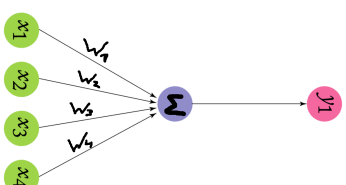
Course Communication

- Questions?
  - On-line chat room: https://mattermost.uio.no/ifi-undervisning/channels/in5550
  - GitHub issues: https://github.uio.no/in5550/2025/issues
  - email: in5550-help@ifi.uio.no reaches all course staff.
- submissions, solutions, etc. on the GitHub instance at UiO.
- Messages:
  - Read (or forward) your UiO email regularly;
  - Check the course page and schedule regularly;
  - Participate in the discussions on Mattermost and GitHub

Next week (lecture recording available this Thursday)

Foundations

- Introducing mathematical notation for describing neural and not-neural classifiers.
- From linear regression to feed-forward networks and multi-layer perceptrons.
- Linear vs. non-linear decision boundaries



References I

Anderson, J. A. and Rosenfeld, E. (1988).
Neurocomputing: Foundations of Research.
MIT Press.

Goldberg, Y. (2017).
Neural network methods for natural language processing.
Synthesis Lectures on Human Language Technologies, 10(1):1–309.

Shen, J., Pang, R., Weiss, R. J., Schuster, M., Jaitly, N., Yang, Z., Chen, Z., Zhang, Y., Wang, Y., Skerrv-Ryan, R., Saurous, R. A., Agiomvrgiannakis, Y., and Wu, Y. (2018).
Natural TTS synthesis by conditioning Wavenet on MEL spectrogram predictions.
In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 4779–4783.