

# ML4NLP 1

Tutorial 14: Exam preparation  
16. December 2024

# Schedule & Admin

- This week
  - Exam preparation
- Feedback for Ex. 5 coming soon...
- Peer Review for Ex.6 starts today (deadline: 06.01.2024)

# General Exam Style

- A4 cheat sheet allowed (both sides handwritten, one side if printed)
- Exam questions: Mostly Conceptual, basic calculations or code snippets can come up
- Primarily open ended questions on the Lecture Content, some True/False and multiple choice (~15-25% of points)
- No negative grading
  - It's always better to answer something (don't go off topic, make short and precise guesses)
- Elaborate use case scenarios questions are to be expected
  - (E.G) You are an NLP Engineer and you are asked to design system X using the following material Y.
- Questions about code discussed in tutorials might come up...

# Exam: Common Topics

- Historical development of NLP
- High-level questions about challenges in NLP
- General ML concepts
- Word embeddings
- Clustering
- Multitask Learning
- Transfer learning
- Neural Network model types:
  - FFNNs
  - RNNs
  - CNNs
  - Transformers
- Pre-trained LMs
- Pre-trained / Chat-tuned LLMs

# Examples of High-level Questions

1. What makes ML for NLP challenging? Mention two important and general sources/types of difficulties of NLP. For each type, mention one current technique that helps to alleviate it and why it roughly works.
2. What are two instances of unsupervised learning techniques used in NLP.

# ML

Joe claims that for a complex machine learning task, where various features need to be considered, the following is true: training a single **linear regression model** with 500 features is equivalent to building an ensemble of 500 models each with a single feature.

Would you **agree**, yes or no? Add a **short justification** for your answer.

# Word Embeddings

1. What are the commonalities / differences between the following?
  - [word2ec](#)
  - [Glove](#)
  - [ELMo](#)
  - [BERT](#)
2. Name two advantages of contextualized word embeddings over static word embeddings

# GPT

1. How would you explain to a layperson the important ideas behind GPT3 in your own words?
  - 176B parameters and was **trained** on web-crawled text, books and code
  - pretrained using **generative language modeling** - learns to predict the next word in a piece of text
  - words are not “words” per se, but “**subtokens**” - allows GPT to represent and produce also rare words with a limited vocabulary
  - It’s architecture is based on a **transformer decoder** that process a **fixed amount of subtokens** in a single forward pass.
  - A GPT-like model *can* be applied to any text-based task by framing it as a **prompt** that elicits the correct answer as a valid continuation.



# Language Modelling Objectives

1. What's the difference between the following learning objectives?
  - Masked Language Modelling
  - Generative Language Modelling
  - Translational Language Modelling

# Language Modelling Objectives ctd.

The XLM model called xlm-mlm-tlm-xnli15-1024 is pretrained with which of the following tasks:

- ☐ Masked Language Modelling
- ☐ Generative Language Modelling
- ☐ Translational Language Modelling
- ☐ Next Sentence Prediction

GPT and ELMO have in common that they use generative language modeling for pretraining.

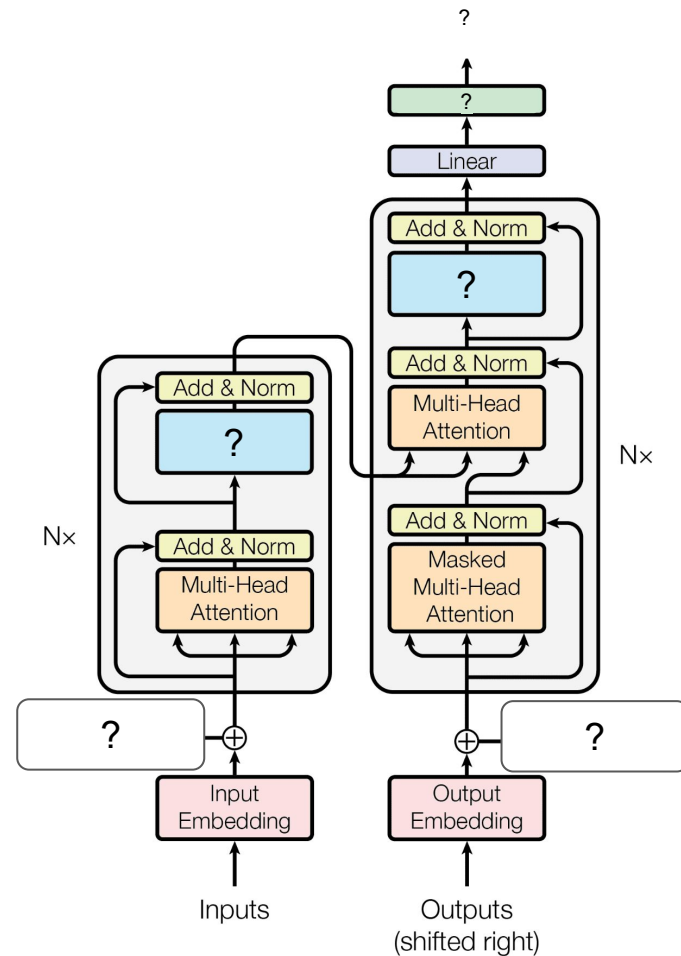
- ☐ True
- ☐ False

GPT and BERT have in common that they use generative language modeling for pretraining.

- ☐ True
- ☐ False

# Transformer

1. Locate the encoder and the decoder in the figure:
2. Label the parts of the architecture marked with ?
3. What are the differences between the following:
  - a. Self-attention in the encoder
  - b. Cross-attention
  - c. Self-attention in the decoder



# Pretrained LMs

1. Given the following parameter dictionary for a mini LM, what is the total parameter count of the model? (keys indicate the named params, values indicate their sizes)

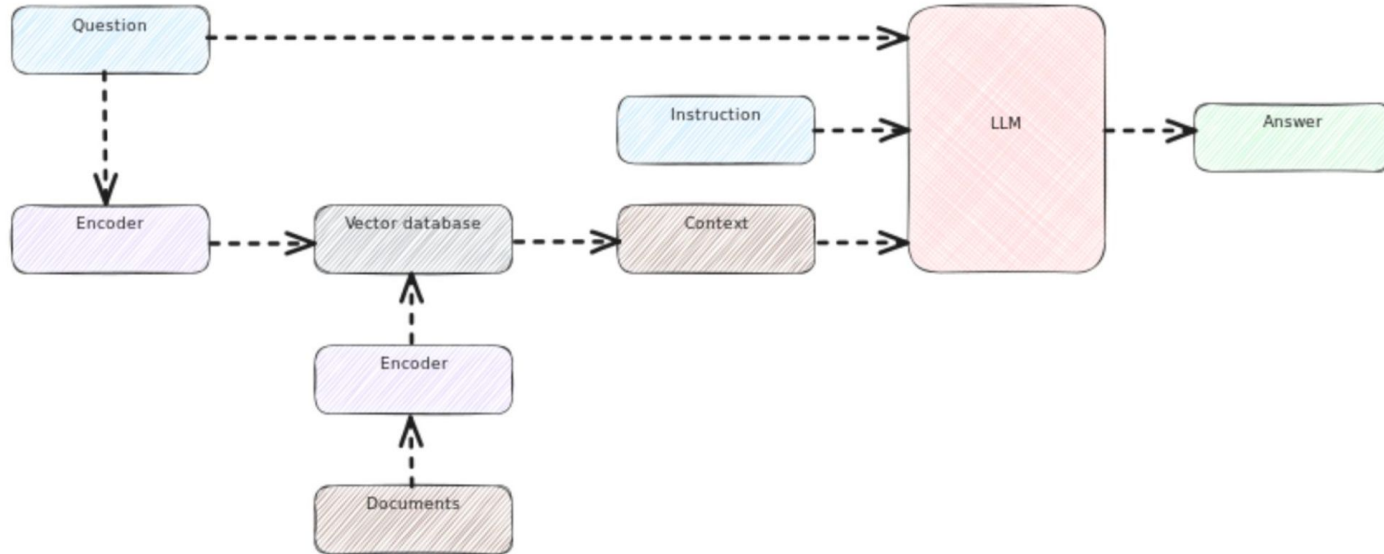
```
m = {'blocks': [  
    {'attn': {'c_attn': {'b': [20], 'w': [5, 20]},  
                'c_proj': {'b': [5], 'w': [5, 5]}},  
    {'ln_1': {'b': [5], 'g': [5]},  
    {'ln_2': {'b': [5], 'g': [5]},  
    {'mlp': {'c_fc': {'b': [30], 'w': [5, 30]},  
              'c_proj': {'b': [5], 'w': [30, 5]}}}  
],  
    'ln_f': {'b': [5], 'g': [5]},  
    'wpe': [20, 5],  
    'wte': [50, 5]  
}]
```

# LLMs

1. What is the main difference between GPT-3 (the base model described in [Brown et al., 2020](#)) and [GPT-3.5-Turbo](#) (the model that was presumably powering the first instances of ChatGPT)?
2. The management has decided to get on the Generative AI bandwagon and want LLMs integrated into their product offering. However, they are concerned with data privacy and will not allow the use of paywalled API models. What would you suggest as a way forward? What are the pros and cons?

# LLM Inference

1. What type of technique does the following image illustrate? What issue does it try to fix?



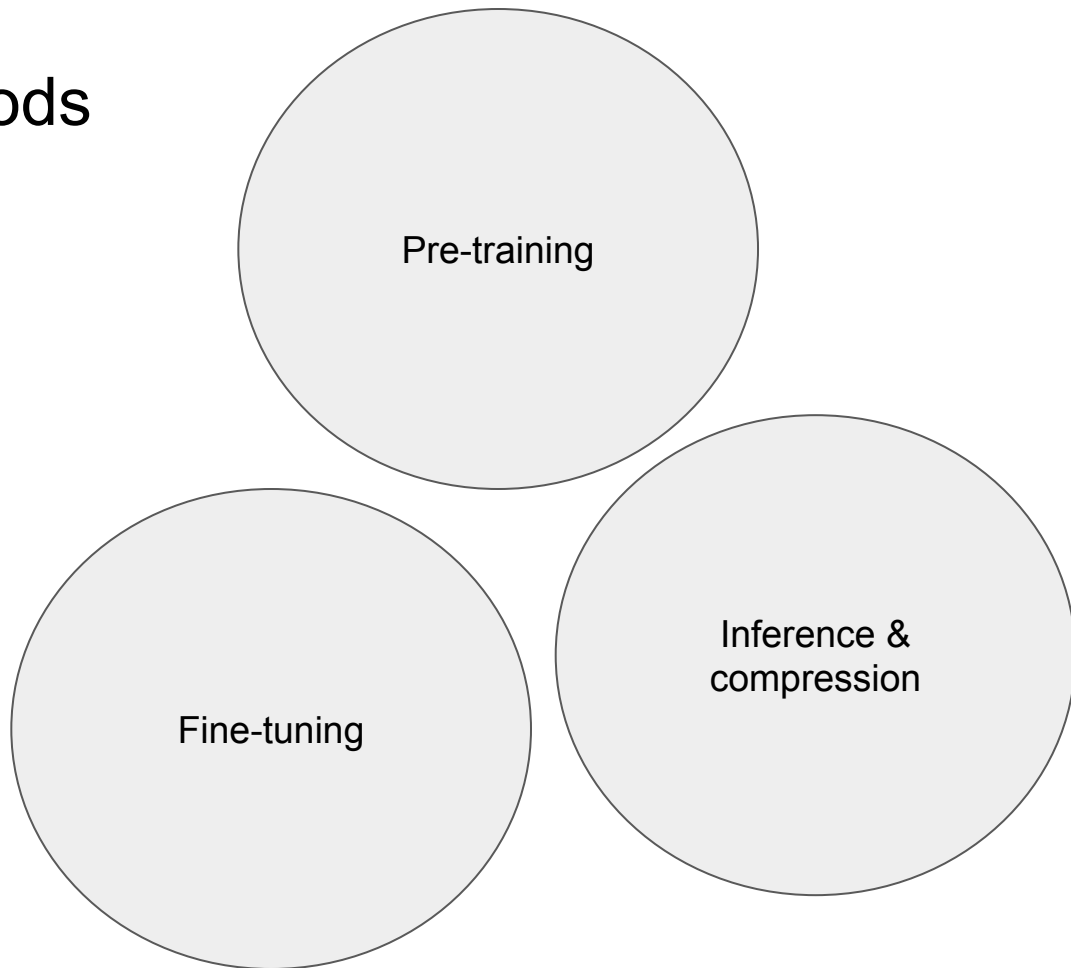
# Parameter-efficient methods

Assign the following models/techniques to their umbrella term(s) (can be more than 1).

- GPT-3
- Zero-shot learning
- Distillation
- FLAN
- Adapters
- Pre-training

Recommended reading:

[Treviso et al. 2022: Efficient Methods for Natural Language Processing: A Survey](#)



# Parameter-efficient methods

1. Name one advantage and one disadvantage of adapters.
2. What's the difference between prompt-tuning and prefix-tuning? And which of the two is more parameter-efficient?



# Blackbox NLP

1. What are some techniques/approaches that we can use to take a look into the black box?
2. Name a specific example where the inner workings of the blackbox would be worthwhile/important to investigate.

# Clustering

1. What is the difference between clustering and classification?
2. Given the primary goals of clustering, what concrete evaluation measurements can be used to assess the quality of topic modeling?

# Further Involvement with CL / NLP

Interesting classes (that are also applicable for Informatics, CL people know what exists):

## Fall Semester:

- Advanced Techniques of Machine Translation
- Machine Learning for Natural Language Processing 1

## Spring semester:

- Advanced Machine Learning
- Machine Learning for Natural Language Processing 2

## Open Master Thesis topics within Computational Linguistics / NLP:

<https://www.cl.uzh.ch/en/studies/studies-BA-MA/teaching/BaMasterArbeit.html>