

Understanding and Training Language Models: Introduction

Erik-Jan Senn

Faculty of Mathematics and Statistics, University of St. Gallen

CSH Autumn School at University of Hohenheim
September/October 2024

Roadmap

Hello

Course Structure

Language Modelling Pipeline

Introduction

Hello

Hello from you!

- ▶ Who are you?
- ▶ What do you want to learn here?
- ▶ How do you (want to) use Language Models?
- ▶ What experiences do you have in:
 - ▶ Statistical/machine learning
 - ▶ Language processing
 - ▶ Coding

Hello from me!

- ▶ Who are you?
A PhD Candidate in Econometrics at the University of St. Gallen ([Website](#)).
- ▶ What do you want to learn here?
Teach to learn, learn to teach. :)
- ▶ How do you (want to) use Language Models?
Sentiment analysis for financial forecasting (tweets), accounting fraud detection (accounting filings).
- ▶ What experiences do you have in:
 - ▶ Statistical/machine learning *In my PhD, I broadly speaking apply these tools to prediction problems in economics/finance.*
 - ▶ Language processing *I apply and work on LLMs in the context of research projects in sentiment analysis for finance.*
 - ▶ Coding *No a professional. Some years of experience in R/python mainly for data and modelling work. I write ugly code that sometimes works. No expert in torch / tensorflow.*

Introduction

Course Structure

Summary

What will you learn?

- ▶ Understand and implement the components of the language modeling pipeline,
- ▶ use and train foundation models,
- ▶ apply and adapt pretrained models for specific supervised tasks (mainly sentiment analysis).

Summary

What will you learn?

- ▶ Understand and implement the components of the language modeling pipeline,
- ▶ use and train foundation models,
- ▶ apply and adapt pretrained models for specific supervised tasks (mainly sentiment analysis).

What do you have to do?

- ▶ Attend,
- ▶ run and write code (notebooks with exercises, project),
- ▶ participate actively (questions, comments).

Topics covered

1. Text Preprocessing
2. Statistical Learning and Neural Networks
3. Self-attention and Transformers
4. Training Foundation Language Models
5. Task-specific training and Fine-Tuning

Planned Schedule

Day 1: 30.09.2024, 9h00 - 17h30:

| | |
|---------------|---|
| 9h00 - 10h30 | Lecture: Introduction, language modeling pipeline |
| 10h30 - 11h00 | Coffee Break |
| 11h00 - 12h30 | Lecture: Text preprocessing |
| 12h30 - 13h30 | Lunch Break |
| 13h30 - 15h30 | Exercise: Text preprocessing |
| 15h30 - 16h00 | Coffee Break |
| 16h00 - 17h30 | Lecture: Statistical Learning and Neural Networks |

Planned Schedule

Day 2: 1.10.2024, 2024, 9h00 - 17h30:

| | |
|---------------|---|
| 9h00 - 10h30 | Lecture and Exercise: Training a neural network |
| 10h30 - 11h00 | Coffee Break |
| 11h00 - 12h30 | Lecture and Exercise: Self-attention and transformers |
| 12h30 - 13h30 | Lunch Break |
| 13h30 - 15h30 | Lecture: Foundation models |
| 15h30 - 16h00 | Coffee Break |
| 16h00 - 17h30 | Exercise: Training foundation models |

Planned Schedule

Day 3: 2.10.2024, 2024, 9h00 - 17h30:

| | |
|---------------|--|
| 9h00 - 10h30 | Lecture: Task-specific training and fine-tuning |
| 10h30 - 11h00 | Coffee Break |
| 11h00 - 12h30 | Exercise: Task-specific training and fine-tuning |
| 12h30 - 13h30 | Lunch Break |
| 13h30 - 15h30 | Exercise: Sentiment analysis project |
| 15h30 - 16h00 | Coffee Break |
| 16h00 - 17h30 | Lecture and Exercise: Generative language models, conclusion |

Resources

1. Course Github https://aidaho-edu.uni-hohenheim.de/gitlab/eriksenn/llm_class_public.
Copy/fork under different name. Pull daily to get updated material.
 - ▶ Slides
 - ▶ Notebooks (including exercises)
 - ▶ Data
2. JupyterHub Hohenheim <https://aidaho-edu.uni-hohenheim.de/CSHAS2024>.
Our computing environment. Use GPUs only when necessary.
3. Main Reference: Book and codes for "Build A Large Language Model (From Scratch)" by [Sebastian Raschka](#).
[Codes \(open source\)](#), [Book \(to purchase, not required\)](#) (Raschka, 2024), [Figures](#).
4. Other nice references (that I partly rely on):
 - ▶ Deep learning class by Alfredo Canziani [course github](#)
 - ▶ Notebooks for neural networks from my great colleague [Jonathan Chassot](#) from [this bachelor course](#)
 - ▶ Deep learning notebooks by Andrej Karpathy [github](#)
 - ▶ Nice deep learning [visualizations](#).

Please let me know if you have any [feedback](#), found some typos, ...

Introduction

Introduction to Natural Language Processing

What are language models capable of?

Example from *BERT* for review classification:

1. "This movie was fantastic! I loved it.",
2. "The food was terrible, I will never come back."
3. "The service was just okay, nothing special."

What are language models capable of?

Example from *BERT* for review classification:

1. "This movie was fantastic! I loved it.",
2. "The food was terrible, I will never come back."
3. "The service was just okay, nothing special."

```
1  
Sentiment: 5 stars  
  
2  
Sentiment: 1 star  
  
3  
Sentiment: 3 stars
```


What are language models capable of?

Example from *ChatGPT 4o* for text generation:

I need to fill out my personal introduction in the provided latex slide.

I will upload my CV. Can you fill out my personal introduction and give me the latex code?

What are language models capable of?

Example from *ChatGPT 4o* for text generation:

I need to fill out my personal introduction in the provided latex slide.

I will upload my CV. Can you fill out my personal introduction and give me the latex code?

```
latex Copy code

\begin{frame}{Hello from Erik-Jan Senn!}
\begin{wideitemize}
  \item Who are you? \\
  My name is Erik-Jan Senn, a PhD candidate in Economics and Finance at the University of
  \item What do you want to learn here? \\
  I aim to deepen my understanding of how machine learning models can be applied to econ
  \item How do you (want to) use Language Models? \\
  I use language models for tasks such as financial sentiment analysis, automating parts
  \item What experiences do you have in:
  \begin{wideitemize}
    \item Statistical/machine learning \\
    I have experience in statistical and machine learning techniques, particularly in
    \item Language processing \\
    I work on financial sentiment analysis using text data from various financial sour
    \item Coding \\
    I am proficient in Python and R, with additional experience in MATLAB, SAS, and VB
  \end{wideitemize}
\end{wideitemize}
\end{frame}
```

What are language models capable of?

Example from *Github Copilot* for code (text) generation:

```
# Write a function to find the first k prime numbers
```

```
def
```

What are language models capable of?

Example from *Github Copilot* for code (text) generation:

```
# Write a function to find the first k prime numbers
```

```
def
```

```
# Write a function to find the first k prime numbers
def find_k_primes(k):
    primes = []
    num = 2
    while len(primes) < k:
        is_prime = True
        for prime in primes:
            if num % prime == 0:
                is_prime = False
                break
        if is_prime:
            primes.append(num)
        num += 1
    return primes
```

Natural Language Processing (NLP)

1. Humans communicate using a “natural language” (e.g. English), but machines (computers) use numbers.
2. Lots of relevant information exists (and is stored) in the form of human language, not numbers.

Natural Language Processing (NLP)

1. Humans communicate using a “natural language” (e.g. English), but machines (computers) use numbers.
2. Lots of relevant information exists (and is stored) in the form of human language, not numbers.

Aim of NLP: Enable machines (computers) to **understand and communicate** in human language.

Example Tasks in NLP



Tweet of Elon Musk, 2018

Example NLP tasks in increasing order of difficulty:

- ▶ Does the tweet meet the (old) 140 character limit on twitter?
Yes/No 1/0
- ▶ What is the sentiment of the tweet towards Tesla?
Positive/Neutral/Negative 1/0/-1
- ▶ Analyze why this tweet might be related to stock market manipulation.
{Text output}

Key Features of Text Data

- ▶ Text follows certain syntactic **rules** and a hierarchical structure.
- ▶ Sequential / **time series** structure of sentences / words.
- ▶ There is lots of text **data available!***

Nice for modeling!

Key Features of Text Data

- ▶ **Sparsity**: many words/sequences appear very infrequently (and some very often).
- ▶ **Contextual meaning and ambiguity** of text: e.g. meaning of words depends on surrounding words, and slight changes in wording or word order can change entire meaning of a document.
- ▶ Language is **domain-specific**: different characters, words and rules in different languages, industries, text formats.

Difficult for modeling!

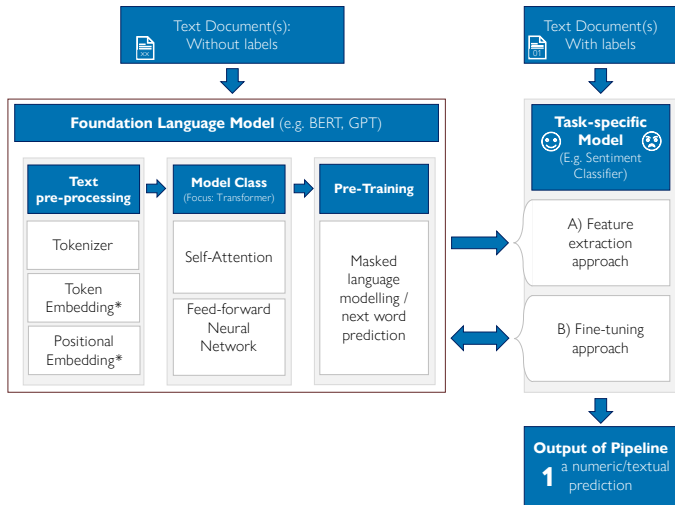
Approaches to NLP

- ▶ Symbolic/Rule-based NLP: **Process** human language **by pre-defined rules**.
 - ▶ Example 1: Identify nouns as words that follow an *the* or *a*.
 - ▶ Example 2: Use pattern matching (regular expressions) to identify dates.

Approaches to NLP

- ▶ Symbolic/Rule-based NLP: **Process** human language **by pre-defined rules**.
 - ▶ Example 1: Identify nouns as words that follow an *the* or *a*.
 - ▶ Example 2: Use pattern matching (regular expressions) to identify dates.
- ▶ Statistical/machine-learning-based NLP: **Learn** human language **from data**.
 - ▶ *Simple* statistical models; e.g. based on word-counting (bag-of-words, *TF-IDF*), *Word2Vec*, ...
 - ▶ Neural (deep learning based) models: **(Large) Language Models (LM/LLMs)**

The Language Modelling Pipeline



Language Modelling Pipeline

Disclaimer

- ▶ We focus on **basic concepts** that appear in most LLMs to improve understanding. However, every LLM is different and we cannot tackle the newest developments in the field.
- ▶ The LLM literature follows using the principle: **“Whatever works in practice is good!”** (My personal opinion.)
We will not focus much on the “Why”, because often there is no theory-guided answer.
- ▶ Mathematical rigour is often sacrificed/ignored for intuition. But please ask if you are interested or the notation is incorrect / confusion.
- ▶ Terminology differs across fields. E.g. Statisticians *estimate* a model, computer scientists *train* it.
- ▶ We focus on non-generative applications of LMs.

Disclaimer (cont'd)

- ▶ Creating well-performing LLMs requires

- (i) a strong model architecture,
- (ii) large amounts of data,
- (iii) and lots of computing power.

Compared to big-tech companies, **we sadly lack (ii) data and (iii) computing resources**. Therefore, in parallel to learning how to build sub-components of an LLM ourselves, we will also learn how to use pretrained LLM components for better performance.

- ▶ We will work with `torch` and `transformers` mainly.
- ▶ First iteration of the course, and fresh material. Happy for feedback of any kind.

Questions ?

References

Sebastian Raschka. *Build A Large Language Model (From Scratch)*. Manning, 2024. ISBN 978-1633437166. URL <https://www.manning.com/books/build-a-large-language-model-from-scratch>.