# Week 11: Textual Analysis & Large Language Model in Accounting and Finance – Part 1

AF3214 Python Programming for Accounting and Finance

Vincent Y. Zhuang, Ph.D.
vincent.zhuang@polyu.edu.hk

School of Accounting and Finance
The Hong Kong Polytechnic University

R508, 8:30 am – 11:20 am, Wednesdays, Semester 2, AY 2024-25

# What is textual analysis?

Extracting meaningful information from text

- This could be as simple as extracting specific words/phrases/sentences
- This could be as complex as extracting latent (hidden) patterns structures within text:
  - ☐ Sentiment
  - ☐ Content
  - ☐ Emotion
  - ☐ Writer characteristics
  - ☐ …
- It is called text mining (in computer science, engineering) or textual analysis (in accounting and finance)

# So What is NLP?

NLP is an emerging field devoted to understanding human language

- NLP stands for Natural Language Processing
- It is a very diverse field within computer science/engineering:
  - Grammar/linguistics
  - Conversations
  - Conversion from audio, images
  - Translation
  - Dictation
  - Generation

# Key difference between text analysis and NLP

## TEXTUAL ANALYSIS GOALS

Derive quality insights from solely the text or words itself without consideration of the Semantics

e.g., frequency of words, length of sentence, and presence or absence of words

## NLP GOALS

Understand the linguistic use and context behind the text with consideration of grammatical structures and semantics.

e.g., semantic meaning of text, sentimental analysis, grammatical structure, automatic text summarization and translation

# Why We Discuss NLP? An example

Let's consider the following situation:

> You have a collection of 1 million sentences, and you want to know which are accounting relevant

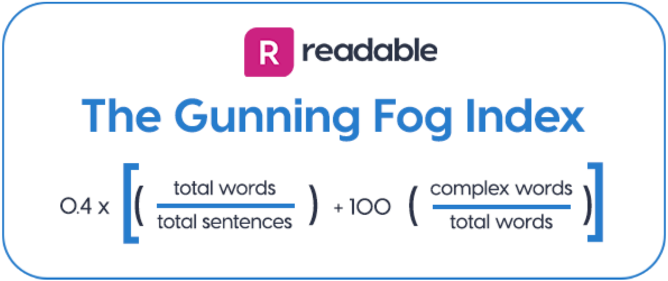
- Without NLP:
  1. Hire a Research Assistant
  2. Use a dictionary: Words/phrases like "earnings," "profitability," "net income" are likely to be in the sentences
- With NLP:
  1. We could associate sentences with outside data to build a classifier (supervised approach)
  2. We could ask an algorithm to learn the structure of all sentences, and then extract the useful part ex post (unsupervised)

# Text Data Studied

- Firms
  - Letters to shareholders
  - Annual and quarterly reports
  - 8-Ks, 10Ks
  - Press releases
  - Conference calls
  - AGM
  - Firm websites
  - X posts
- Investors
- Blog posts
- Social media posts
- …
- Intermediaries
  - Newspaper articles
  - Analyst reports
- Government
  - Comment letters
  - Court cases
- …

# Text Data Readability

- Readability is the ease with which a reader can understand a written text.
- Measure: Gunning Fog Index



readable

**The Gunning Fog Index**

$$0.4 \times \left[ \left( \frac{\text{total words}}{\text{total sentences}} \right) + 100 \left( \frac{\text{complex words}}{\text{total words}} \right) \right]$$

The Gunning Fog formula generates a grade level between 0 and 20. It estimates the education level required to understand the text.
A Gunning Fog score of 6 is easily readable for sixth-graders. Text aimed at the public should aim for a grade level of around 8. Text above a 17 has a graduate level.

(Notes: Complex words are those containing three or more syllables.)

# Textual Analysis Tools

## Python:

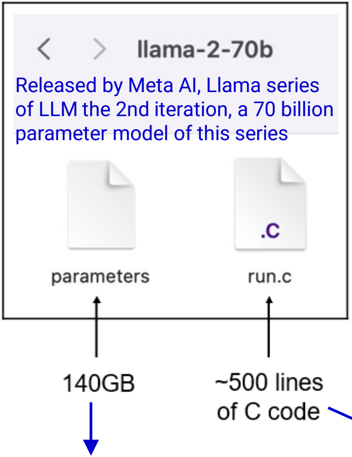
- Text parsing: `spaCy`
- LDA: `gensim`
- Sentiment: `NLTK`, `SpaCy`, or handcode using `Counter()` (super fast)
- Classifiers: `scikit-learn` or `keras` or `pytorch` or `huggingface`
- Other measures: `NLTK`, `spaCy`

## R:

- LDA: `stm`+ `quanteda`+ `convert(dfm,to='stm')`
- Sentiment (dictionary): `tidytext`
- Classifiers: `caret`, `e1071`, or `keras`
- Other measures: Using python is likely better

- Also useful: MALLET, Stanford NLP

# What Is A Large Language Model-LLM?



**llama-2-70b**

Released by Meta AI, Llama series of LLM the 2nd iteration, a 70 billion parameter model of this series

.c

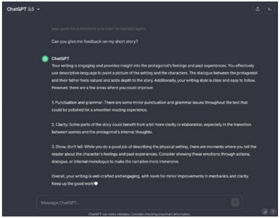
parameters

run.c

140GB

~500 lines of C code

- ➢ Second iteration Llama series of LLM
- ➢ Multiple models for llama2 Series 7b, 13b, 34b, 70b…
- ➢ Most powerful open weights model
- ➢ ChatGPT models are never released, use only through web interface



ChatGPT 3.5

was quiet for a moment and then he started again.

Can you give me feedback on my short story?

**ChatGPT**

Your writing is engaging and provides insight into the protagonist's feelings and past experiences. You effectively use descriptive language to paint a picture of the setting and the characters. The dialogue between the protagonist and their father feels natural and adds depth to the story. Additionally, your writing style is clear and easy to follow. However, there are a few areas where you could improve:

1. Punctuation and grammar: There are some minor punctuation and grammar issues throughout the text that could be polished for a smoother reading experience.

2. Clarity: Some parts of the story could benefit from a bit more clarity or elaboration, especially in the transition between scenes and the protagonist's internal thoughts.

3. Show, don't tell: While you do a good job of describing the physical setting, there are moments where you tell the reader about the character's feelings and past experiences. Consider showing these emotions through actions, dialogue, or internal monologue to make the narrative more immersive.

Overall, your writing is well-crafted and engaging, with room for minor improvements in mechanics and clarity. Keep up the good work!

ChatGPT can make mistakes. Consider checking important information.

weights of parameters of this neural network. 2 bytes (float 16) each, 70b ->140GB. Plus a large list of parameters for that nn

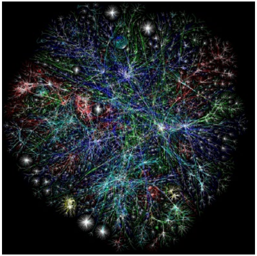
- 500 lines to implement NN architecture and to use parameters to run the model
- No other dependencies, self-contained
- Take two files and compile c code
- Get a binary to point at the parameters, and talk to LLM

# Computational Complexity - Parameters

Computational complexity really comes in when we'd like to get those parameters. What are parameters and where do they come from?

to obtain parameters we need model training

Model inference is quite easy, but model training is computationally very involved process. We can think of it like compressing the internet.



Chunk of the internet,
~10TB of text

imaging collecting tons of text from all kinds of different websites

6,000 GPUs for 12 days for
llama-2-70b, ~$2M, ~1e24 LOPS

procure a GPU cluster for very heavy computational workloads like training of neural networks

parameters.zip

~140GB file

lossy compression of this large chunk of text into a zip file

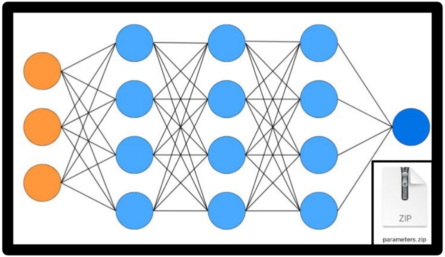
# Computational Complexity – Neural Network

So what are NN doing: Predicts the next word in the sequence

feed in a sequence of words into NN

**cat**
**sat**
**on**
**a**

e.g. context of 4 words



**mat (97%)**

predict next word

- pick a word
- continue feeding it back in
- get the next word
- continue feeding that back in
- iterate this process

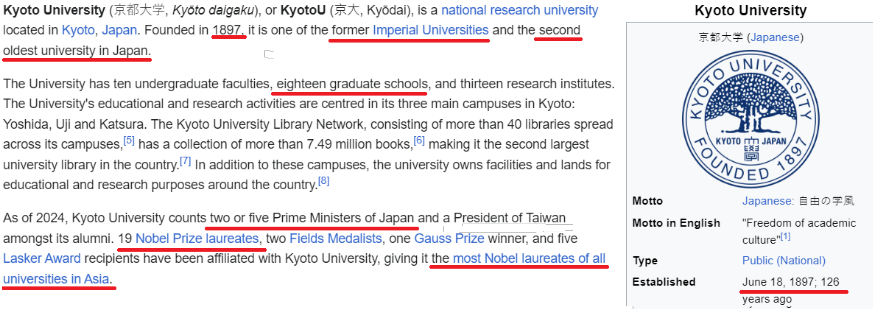
- billions of parameters are dispersed throughout entire NN
- adjust these parameters iteratively to make the network as a whole better at the next word prediction
- we don’t actually really know what these billions of parameters are doing

We can demonstrate mathematically that there's a very close relationship between prediction and compression. So allude NN as kind of training (after Internet compression, because if you can predict next work, then compression)

# Next word prediction

The reason that what you get out of the training is actually quite a magical artifact because:

Next word prediction is a very powerful objective as it forces you to learn a lot about the world inside the parameters of the NN



**Kyoto University** (京都大学, *Kyōto daigaku*), or **KyotoU** (京大, Kyōdai), is a national research university located in Kyoto, Japan. Founded in 1897, it is one of the former Imperial Universities and the second oldest university in Japan.

The University has ten undergraduate faculties, eighteen graduate schools, and thirteen research institutes. The University's educational and research activities are centred in its three main campuses in Kyoto: Yoshida, Uji and Katsura. The Kyoto University Library Network, consisting of more than 40 libraries spread across its campuses,[5] has a collection of more than 7.49 million books,[6] making it the second largest university library in the country.[7] In addition to these campuses, the university owns facilities and lands for educational and research purposes around the country.[8]

As of 2024, Kyoto University counts two or five Prime Ministers of Japan and a President of Taiwan amongst its alumni. 19 Nobel Prize laureates, two Fields Medalists, one Gauss Prize winner, and five Lasker Award recipients have been affiliated with Kyoto University, giving it the most Nobel laureates of all universities in Asia.

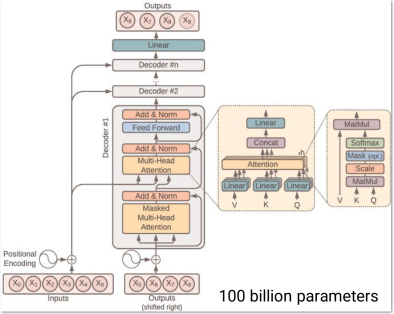
**Kyoto University**

京都大学 (Japanese)

| | |
|---|---|
| **Motto** | Japanese: 自由の学風 |
| **Motto in English** | "Freedom of academic culture"[1] |
| **Type** | Public (National) |
| **Established** | June 18, 1897; 126 years ago |

- think about being the NN and given some amount of words and trying to predict the next word in a sequence. If this is the objective, presumably the parameters have to learn a lot of this knowledge in this task
- you're learning a ton about the world eventually all of this knowledge is being compressed into the parameters.

# How do Models Work?



100 billion parameters

schematic diagram of transformer NN architecture

we know exactly what mathematical operations happen at all the different stages of it.

**but little is known in full detail of parameters…**

- 100 billions of parameters are dispersed through network; we know how to iteratively adjust them (optimize/adjust) to make the network as a whole better at next word prediction.
- we can measure how it works, but we don't really know how these parameters collaborate to do it.

They build and maintain some kind of knowledge database, but it is a bit strange and imperfect:

one dimensional – need to ask from a certain direction

**Recent viral example: "reversal curse"**

Q: "Who is Tom Cruise's mother"?
A: Mary Lee Pfeiffer ✅

Q: "Who is Mary Lee Pfeiffer's son?"
A: I don't know ❌

🤔

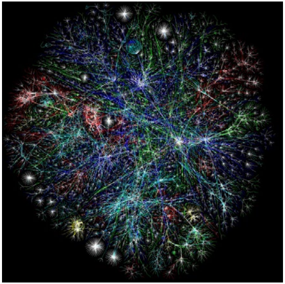
=> think of LLMs as mostly inscrutable artifacts, as opposed to engineering in which we know all the parts, NN comes from long process of optimizations. develop correspondingly sophisticated evaluations.

# Training the Assistant

First stage of training: generating internet document(pre-training stage)
Second stage of training: fine-tuning(where we obtain the assistant model)

- why: a document generators is not very helpful for many tasks
- objective: to generate answers based on the given questions
- solution: an assistant model -> fundamentally through ↓
- process: keep optimization identical(same training), swap the dataset from internet documents with human collected labeled data, then continue training

written by people:



**<USER>**
Can you write a short introduction about the relevance of the term "monopsony" in economics? Please use examples related to potential monopsonies in the labour market and cite relevant research.

**<ASSISTANT>**
"Monopsony" refers to a market structure where there is only one buyer for a particular good or service. In economics, this term is particularly relevant in the labor market, where a monopsony employer has significant power over the wages and working conditions of their employees. The presence of a monopsony can result in lower wages and reduced employment opportunities for workers, as the employer has little incentive to increase wages or provide better working conditions...

human filled ideal response from labeling documents

**~100K conversations**

# Training the Assistant

Low quality, quantity over quality

First stage of training: generating internet document(pre-training stage)
Second stage of training: fine-tuning(where we obtain the assistant model)

High quality, quality over quantity, e.g., 100,000 high quality conversations

We swap out the dataset and we train on these Q&A documents(fine-tuning)

After fine-tuning, we have an assistant model which now subscribes to the form of its new training documents

1st stage: train on internet(knowledge)
2nd stage: fine-tuning(alignment)
change their formatting from internet documents to Q&A documents (a helpful assistant)

**<USER>**

Can you help me with this code? It seems like there is a bug. print(“hello world)

not part of the training set

**<ASSISTANT>**

It looks like you forgot to close the string passed to the function print. You have to add a closing quote to properly terminate the string. Here is the corrected function:

print(“hello world”)

after fine-tuning to answer in a style of a helpful assistant

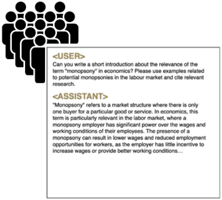
Let me know if I can help with anything else!

# Summary: how to train your GenAI - ChatGPT

every
~year

**Stage 1: Pre-training**

1. Download ~10TB of text.
2. Get a cluster of ~6,000 GPUs.
3. Compress the text into a neural network, pay ~$2M, wait ~12 days.
4. Obtain **base model**.



<USER>
Can you write a short introduction about the relevance of the term "monopsony" in economics? Please use examples related to potential monopsonies in the labour market and cite relevant research.

<ASSISTANT>
"Monopsony" refers to a market structure where there is only one buyer for a particular good or service. In economics, this term is particularly relevant in the labor market, where a monopsony employer has significant power over the wages and working conditions of their employees. The presence of a monopsony can result in lower wages and reduced employment opportunities for workers, as the employer has little incentive to increase wages or provide better working conditions…

every
~week

**Stage 2: Fine-tuning**

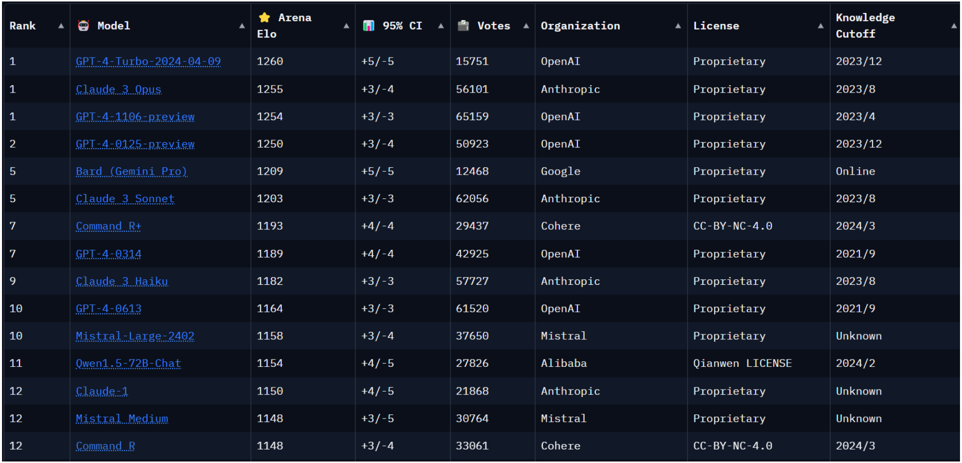
1. Write labeling instructions
2. Hire people, collect 100K high quality ideal Q&A responses, and/or comparisons.
3. Finetune base model on this data, wait ~1 day.
4. Obtain **assistant model**.
5. Run a lot of evaluations.
6. Deploy.
7. Monitor, collect misbehaviors, fix, go to step 1.

llamar-2 models by Meta contain both base models and assistant models

# LLM Leaderboard

LLM Leaderboard from “Chatbot Arena”, managed by UC Berkeley Skylab

As of April, 2024



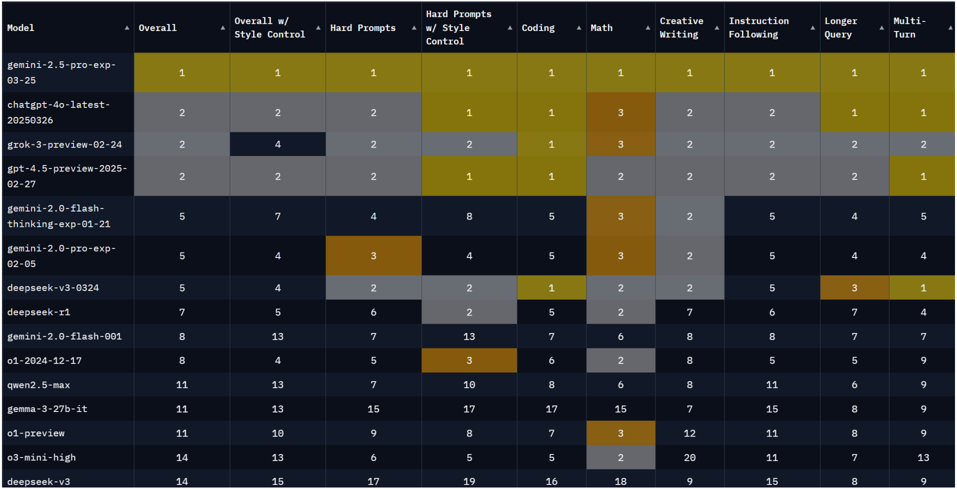
| Rank | Model | Arena Elo | 95% CI | Votes | Organization | License | Knowledge Cutoff |
|---|---|---|---|---|---|---|---|
| 1 | GPT-4-Turbo-2024-04-09 | 1260 | +5/-5 | 15751 | OpenAI | Proprietary | 2023/12 |
| 1 | Claude 3 Opus | 1255 | +3/-4 | 56101 | Anthropic | Proprietary | 2023/8 |
| 1 | GPT-4-1106-preview | 1254 | +3/-3 | 65159 | OpenAI | Proprietary | 2023/4 |
| 2 | GPT-4-0125-preview | 1250 | +3/-4 | 50923 | OpenAI | Proprietary | 2023/12 |
| 5 | Bard (Gemini Pro) | 1209 | +5/-5 | 12468 | Google | Proprietary | Online |
| 5 | Claude 3 Sonnet | 1203 | +3/-3 | 62056 | Anthropic | Proprietary | 2023/8 |
| 7 | Command R+ | 1193 | +4/-4 | 29437 | Cohere | CC-BY-NC-4.0 | 2024/3 |
| 7 | GPT-4-0314 | 1189 | +4/-4 | 42925 | OpenAI | Proprietary | 2021/9 |
| 9 | Claude 3 Haiku | 1182 | +3/-3 | 57727 | Anthropic | Proprietary | 2023/8 |
| 10 | GPT-4-0613 | 1164 | +3/-3 | 61520 | OpenAI | Proprietary | 2021/9 |
| 10 | Mistral-Large-2402 | 1158 | +3/-4 | 37650 | Mistral | Proprietary | Unknown |
| 11 | Qwen1.5-72B-Chat | 1154 | +4/-5 | 27826 | Alibaba | Qianwen LICENSE | 2024/2 |
| 12 | Claude-1 | 1150 | +4/-5 | 21868 | Anthropic | Proprietary | Unknown |
| 12 | Mistral Medium | 1148 | +3/-5 | 30764 | Mistral | Proprietary | Unknown |
| 12 | Command R | 1148 | +3/-4 | 33061 | Cohere | CC-BY-NC-4.0 | 2024/3 |

https://chat.lmsys.org/?leaderboard

# LLM Leaderboard

LLM Leaderboard from “Chatbot Arena”, managed by UC Berkeley Skylab

As of April, 2025



| Model | Overall | Overall w/ Style Control | Hard Prompts | Hard Prompts w/ Style Control | Coding | Math | Creative Writing | Instruction Following | Longer Query | Multi-Turn |
|---|---|---|---|---|---|---|---|---|---|---|
| gemini-2.5-pro-exp-03-25 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| chatgpt-4o-latest-20250326 | 2 | 2 | 2 | 1 | 1 | 3 | 2 | 2 | 1 | 1 |
| grok-3-preview-02-24 | 2 | 4 | 2 | 2 | 1 | 3 | 2 | 2 | 2 | 2 |
| gpt-4.5-preview-2025-02-27 | 2 | 2 | 2 | 1 | 1 | 2 | 2 | 2 | 2 | 1 |
| gemini-2.0-flash-thinking-exp-01-21 | 5 | 7 | 4 | 8 | 5 | 3 | 2 | 5 | 4 | 5 |
| gemini-2.0-pro-exp-02-05 | 5 | 4 | 3 | 4 | 5 | 3 | 2 | 5 | 4 | 4 |
| deepseek-v3-0324 | 5 | 4 | 2 | 2 | 1 | 2 | 2 | 5 | 3 | 1 |
| deepseek-r1 | 7 | 5 | 6 | 2 | 5 | 2 | 7 | 6 | 7 | 4 |
| gemini-2.0-flash-001 | 8 | 13 | 7 | 13 | 7 | 6 | 8 | 8 | 7 | 7 |
| o1-2024-12-17 | 8 | 4 | 5 | 3 | 6 | 2 | 8 | 5 | 5 | 9 |
| qwen2.5-max | 11 | 13 | 7 | 10 | 8 | 6 | 8 | 11 | 6 | 9 |
| gemma-3-27b-it | 11 | 13 | 15 | 17 | 17 | 15 | 7 | 15 | 8 | 9 |
| o1-preview | 11 | 10 | 9 | 8 | 7 | 3 | 12 | 11 | 8 | 9 |
| o3-mini-high | 14 | 13 | 6 | 5 | 5 | 2 | 20 | 11 | 7 | 13 |
| deepseek-v3 | 14 | 15 | 17 | 19 | 16 | 18 | 9 | 15 | 8 | 9 |

https://chat.lmsys.org/?leaderboard

# The End