

# NLP for me

PWYC Microcourse in Natural Language Processing  
October 2024

**Part 1 – Course Overview & Introduction to NLP**  
**Monday, October 7th, 2024**



[nlpfor.me](https://nlpfor.me)

*NLP from scratch* 

# Agenda

**01**

**Welcome & Course Overview**

**02**

**Introduction to NLP**

**03**

**The Toolbox**

**04**

**First Steps in NLP (Hands-on)**

**05**

**Conclusion**

# Welcome!

I'm glad that you've taken an interest in the course. I hope that you will find it valuable as a resource.

The course will cover the fundamentals of natural language processing (NLP), introducing the learner to concepts, tools, and techniques for working with language and machine learning.

While there are technical bits and we will work hands-on in code, this is not intended to be a deep comprehensive look at applying NLP techniques, but rather a place to begin for those unfamiliar with the field.

Let's get started.



**Myles Harrison,**  
AI Consultant & Trainer

# Housekeeping



Camera on if comfortable doing so



Stay muted unless speaking



Be professional



Materials will be shared after the meeting

# Introductions

Let's get (very quickly) introduced! If you're comfortable doing so, please take a minute to share:

- Who you are
- Where you are located
- Professional background / current role & company
- Goals and interest in NLP & AI



# Course Overview

The course will run 5 weeks from Monday, October 5th to Monday, November 4th, 2024.

Course sessions are 7-10 PM EST on Monday evenings.

Office hours are bookable by appointment (each meeting is a 15 minute timeslot) at [nlpfromscratch.com/officehours](http://nlpfromscratch.com/officehours)



**COURSE  
SESSIONS**














Mondays 7-10 PM EST







**OFFICE HOURS**

Wed, Fri 12-1 PM EST

# OCTOBER 2024

SUN	MON	TUE	WED	THU	FRI	SAT
29	30	1	2 	3	4 	5
6	7 	8	9 	10	11 	12
13	14 	15	16 	17	18 	19
20	21 	22	23 	24	25 	26
27	28 	29	30 	31	1	2

# NOVEMBER 2024

SUN	MON	TUE	WED	THU	FRI	SAT
27	28	29	30	31	1 	2
3	4 	5	6 	7	8 	9

# Content & Delivery

The course will span 5 in-person sessions of 3 hours each, covering the topics in the curriculum show on the right.

Course sessions and office hours will be held online through Google Meet.

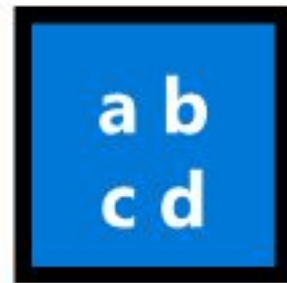
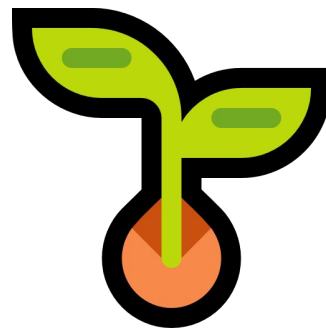
Slides will be provided in PDF format and code in Jupyter notebooks which can either be run locally or through Google Colab.

- 1 Introduction to NLP
- 2 Acquiring & Preprocessing Text
- 3 Machine Learning and Sentiment
- 4 Unsupervised Methods for NLP
- 5 Deep Learning for Natural Language

# Introduction to NLP

You are here. We will introduce the course, the history and current state of natural language processing.

We will also dive into some of the tools and fundamentals for working with natural language in code.





# Acquiring and Preprocessing Text

This refers to the where and how of getting text data, and also methods and techniques for preparing it for whatever task need be accomplished.

Since all NLP tasks require text data and it to be processed beforehand, this is a foundational area.

These will be the topics of Part 2.

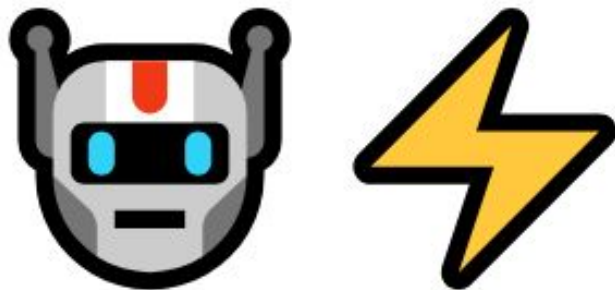


# Machine Learning and Sentiment

Machine learning is the application of statistical methods and algorithms applied to data in order to find patterns, solve problems, or perform tasks.

Sentiment analysis is a subdomain of natural language processing concerned with the emotional tone or content of text.

These topics will be covered together in Part 3 with an applied example.



# Unsupervised Methods for NLP

This type of machine learning is not given specific labelling or prediction tasks, and instead works by finding patterns in the data.

Unsupervised learning is very important for making sense of large bodies of text, and can also be used for transformation of data before applying other machine learning methods.

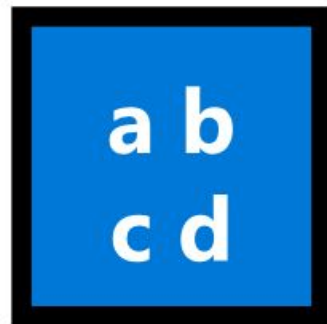
We will cover unsupervised methods including topic modeling and word embeddings in the Part 4.



# Deep Learning for Natural Language

Also known as neural networks, this type of machine learning seeks to emulate how the human brain functions, and represents the state of the art for nearly all NLP tasks.

We will introduce the fundamentals of deep learning and move into its applications to language in the final section.




# Pricing & Payment

This course is offered on a Pay-What-You-Can (PWYC) basis.

You may pay any amount for the course (including \$0), based on what you are able to comfortably afford and that you feel the course is worth.

I would appreciate your support in my developing the course and future content.

You may pay at any time during the course or after the course concludes.

*NLP from scratch* 

Pay NLP from scratch

CA\$0.00

+ tax ⓘ



NLP for me - PWYC

CA\$0.00

NLPfor.me is an online Pay-What-You-Can course in NLP.  
You may pay as much as you're comfortably able and... ▼

Subtotal

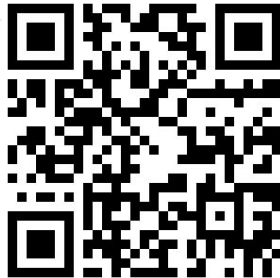
CA\$0.00

Tax ⓘ

CA\$0.00

Total due

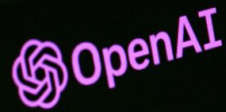
[nlpfromscratch.com/pwyc](https://nlpfromscratch.com/pwyc)





# Introduction to NLP





# What's the deal with this ChatGPT thing?

ChatGPT is an example of a *large language model (LLM)*, a type of deep learning model trained with hundreds of millions or billions of parameters on very large bodies of text. Large language models currently represent the state of the art in NLP.

While we're here, ChatGPT is not sentient, nor is it an example of an Artificial General Intelligence (AGI). Let's take a step back...

## ChatGPT: Optimizing Language Models for Dialogue

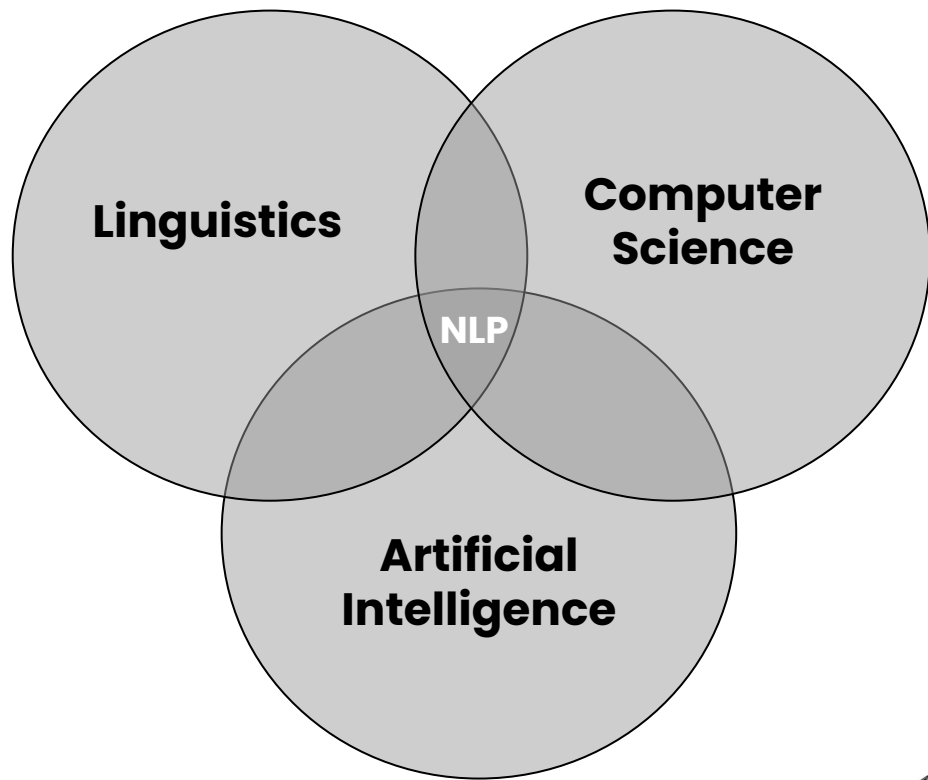
We've trained a model called ChatGPT which interacts in a conversational way. The dialogue format makes it possible for ChatGPT to answer followup questions, admit its mistakes, challenge incorrect premises, and reject inappropriate requests. ChatGPT is a sibling model to InstructGPT, which is trained to follow an instruction in a prompt and provide a response.

# What is Natural Language Processing?

*Natural language processing* lies at the intersection of the domains of linguistics, computer science, and artificial intelligence.

Though the term *processing* usually refers specifically to altering and preparing data, in the domain of AI, NLP is often used to refer more generally to any language problem, including those of applying machine learning (ML) to language, since these still require processing text data beforehand.

Large language models (LLMs) currently represent the state-of-the-art in natural language processing.





# Areas of NLP

The field of NLP can be broken down into high level areas and associated tasks, as non-exhaustively shown here.

Some areas are highly specialized and far beyond the scope of this course.



Document  
Classification



Natural Language  
Generation



Named Entity  
Recognition



Machine  
Translation



Speech  
Recognition



Sentiment  
Analysis



Conversational  
Systems

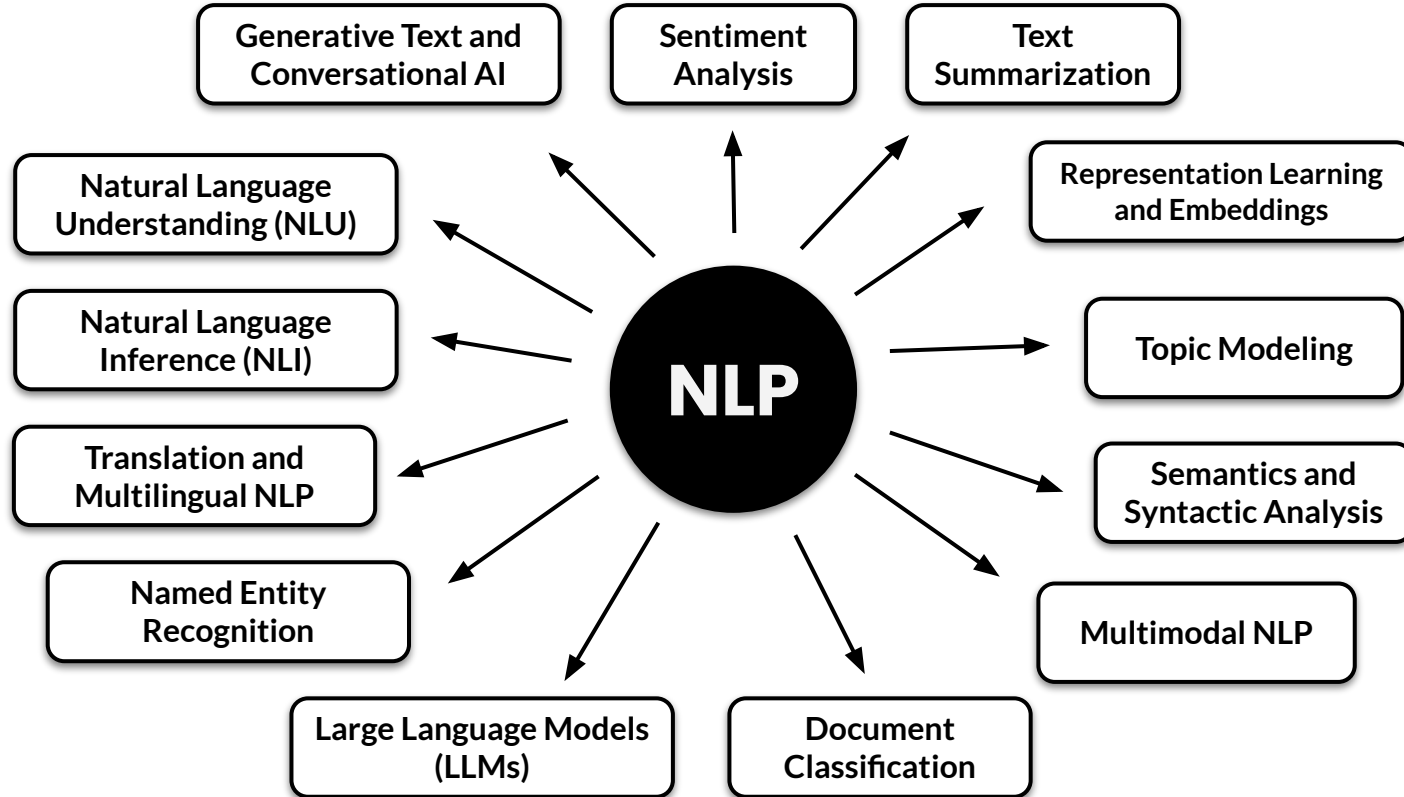


Text to  
Speech

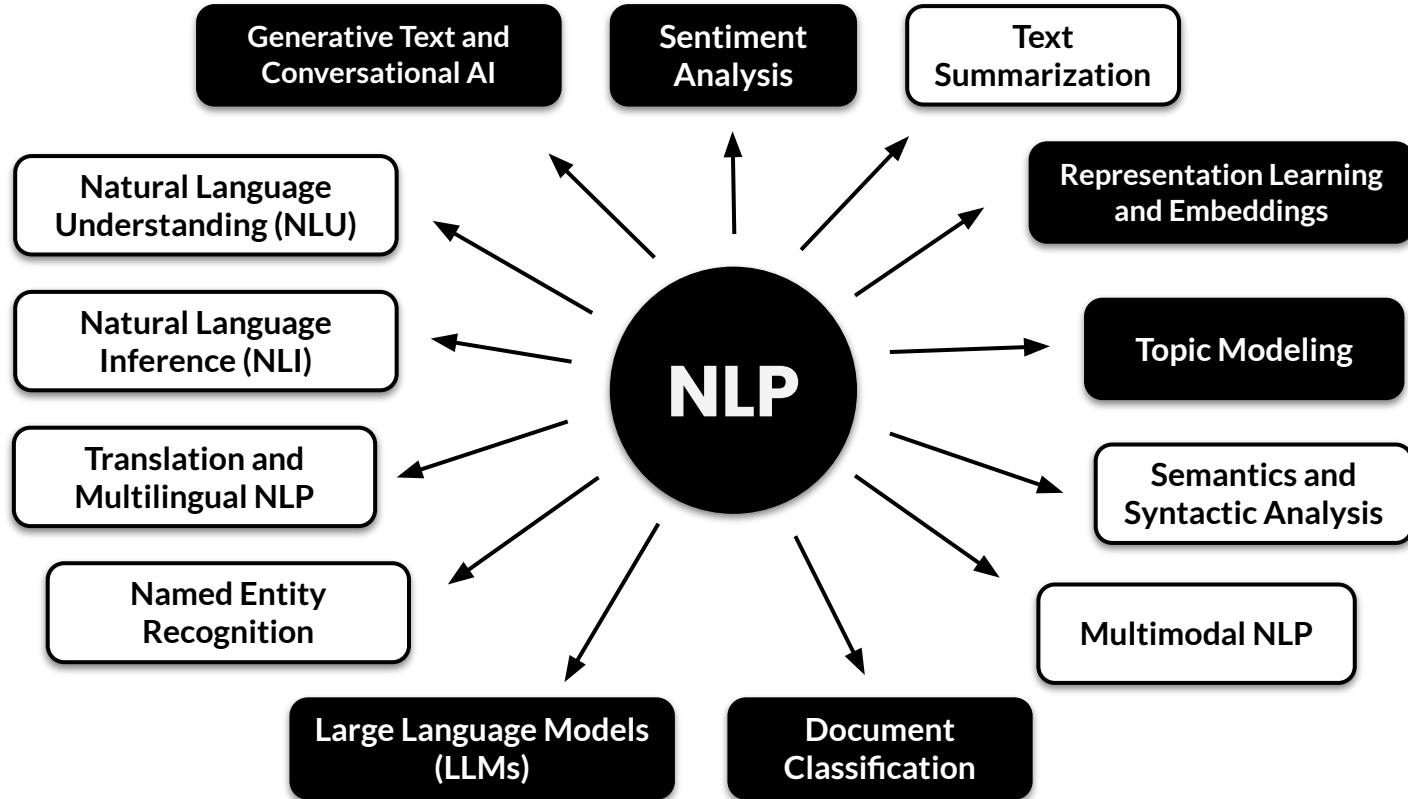


Document  
Summarization

# Common NLP Tasks & Domain Areas



# Common NLP Tasks & Domain Areas



# Applications of NLP

Some examples of use cases for natural language processing and machine learning for specific industry verticals are provided here.



## Finance

Summarizing earnings reports, financial statements, filings, etc.



## Retail

Generative models for copywriting automation



## Medicine

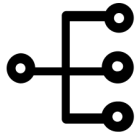
Categorizing and classifying free-form clinical notes



## Media

Automated captioning of television and films

# A Brief History of NLP (according to Wikipedia)



## **Symbolic**

(1950's- 1970's)

Rules-based methods for language tasks such as translation and conversation.



## **Statistical / ML**

(1980's- 2000's)

Advent of statistical techniques and application of machine learning.



## **Neural**

(2000's - Present)

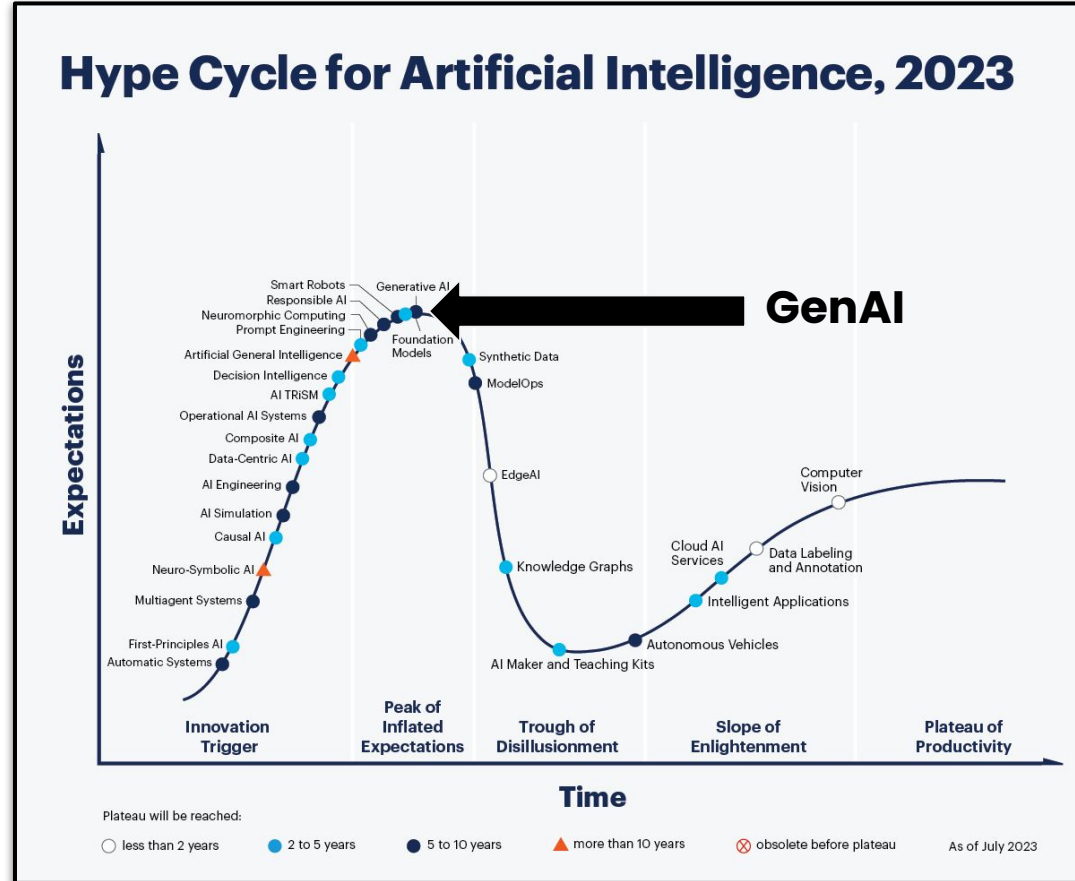
Breakthroughs in deep learning leading to rapid advances in the field up to today.


# The New AI Boom

The AI boom or AI spring is an ongoing period of rapid progress in the field of artificial intelligence (AI) that started in the late 2010s before gaining international prominence in the early 2020s.

Examples include protein folding prediction led by Google DeepMind and generative AI applications developed by OpenAI.

[en.wikipedia.org/wiki/AI\\_boom](https://en.wikipedia.org/wiki/AI_boom)



NLP from scratch 



**The Toolbox**

# Tools of the Trade



Weapon of  
choice



Structured data  
manipulation  
and processing

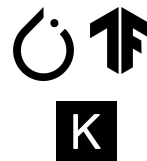


Machine  
learning and  
text processing



Notebooks and  
reproducibility

**spaCy** Industrial  
strength NLP



Deep learning

**NLTK** Fundamentals  
in base python

 **GENSIM** Topic  
modeling  
topic modelling for humans

Large language models



**Hugging Face**



# Tools of the Trade



Weapon of  
choice



Structured data  
manipulation  
and processing

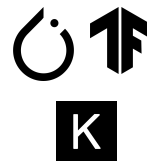


Machine  
learning and  
text processing



Notebooks and  
reproducibility

**spaCy** Industrial  
strength NLP



Deep learning

**NLTK** Fundamentals  
in base python



Topic  
modeling

Large language models



**Hugging Face**

# Installing Python

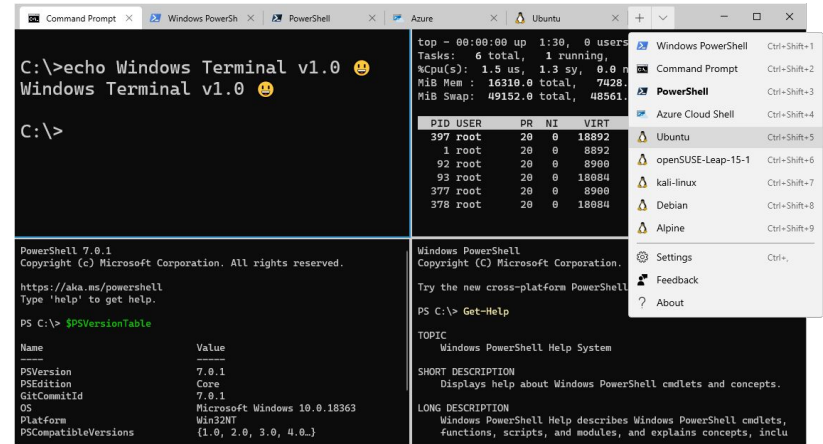
- Installing python on Windows, Mac, or Linux is straightforward.
- The former two do not have installed by default, and can be done by navigating to [python.org](https://python.org), downloading the appropriate installer and following the directions.
- Most Linux distributions already have python installed by default, but may not have **pip** (python's package manager) which may require installation



<https://youtu.be/zyf16rq1R9U>

# Working with the terminal

- On a Windows System, the terminal can be accessed either through the Command Prompt or Powershell
- Working on the terminal in Windows, it is now recommended to use Windows Terminal from the Microsoft Store which allows working with tabs, multiple different types of shells, and using tools such as SSH and Azure Cloud Shell
- On a Mac and Linux systems, terminal is installed by default and can be run through the dock or applications menu



Windows Terminal

# Installing packages with `pip`

- `pip` is python's package manager: handles installation and management of Python libraries ("app store" for libraries)
- Easy to use as only requires a single command for installing, upgrading, and uninstalling packages:  
`pip install <packagename>`
- Packages are installed from PyPI, the Python Package Index which hosts over 300,000 libraries.
- For this course, we will install the "data science stack":  
`pip install numpy pandas matplotlib  
scikit-learn`



# The Data Science Stack in Python



**NumPy**

[numpy.org](http://numpy.org)

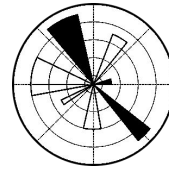
Numeric  
computing in  
python



**Pandas**

[pandas.pydata.org](http://pandas.pydata.org)

Structured data  
manipulation &  
analysis



**Matplotlib**

[matplotlib.org](http://matplotlib.org)

Data  
visualization



**sklearn**

[scikit-learn.org](http://scikit-learn.org)

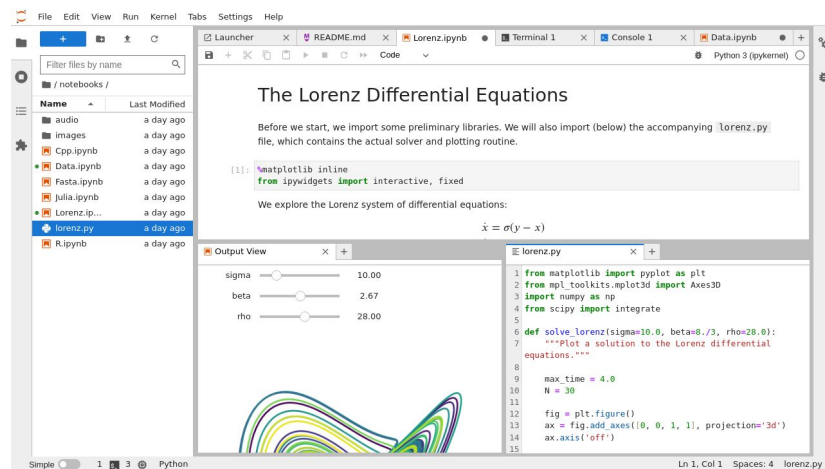
Machine  
Learning

# Installing Jupyter

One of the standard tools for doing data science is *Jupyter* - an open source notebook environment for developing code. Jupyter runs in the browser from a local web server, and allows combining code, code outputs, and documentation (in a language called markdown) to create shareable and interpretable documents for development and analysis.

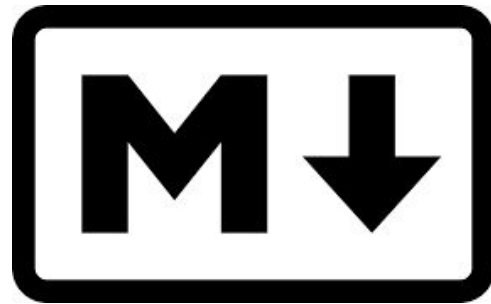
Opinions on Jupyter vary (most developers hate it, most data scientists love it), but it is now a standard component of doing data science work (including NLP).

Jupyter notebooks have evolved in *Jupyterlab*, which can be installed through `pip`, python's package manager: `pip install jupyterlab`



# Markdown Primer

- Markdown is a lightweight markup language for creating formatted text, making documentation, reports, and presentations more readable and professional
- An advantage of markdown is that it can be written entirely in plain text; formatting is included as raw text and then rendered in an application or platform that renders markdown
- It is widely used both in popular tools and platforms (e.g. Jupyter notebooks, Github, Reddit, Stackoverflow, even ChatGPT!)
- Commonly used formatting includes hash symbol for headers (# = H1, ## = H2, etc.), single asterisk for *italics*, and double asterisks for **bold**.



- # **Headings start with a hash**
  - Use asterisks for *italics*\*
  - Use two asterisks for **bold**\*\*
  - Use colons to `::highlight::`

# Installing a code editor

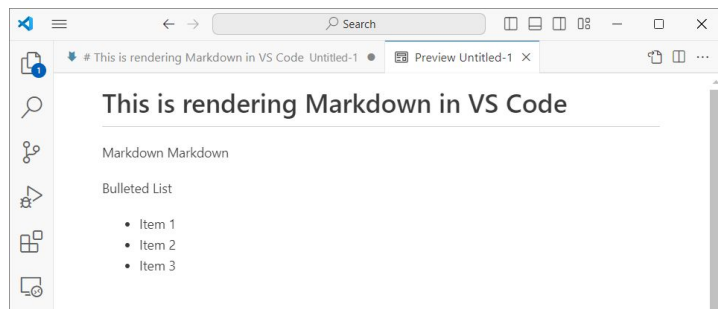
- While Jupyter is fine for prototyping, and the standard tool for data science and AI work, having a good code editor for development (e.g. `.py`) files is important
- Visual Studio Code (VSCode) from Microsoft was released in 2015 and has become widely used and standard
- Lots of features (extensions, git & docker integration, etc.) while remaining (relatively) lightweight
- Disable telemetry 😞  
[https://code.visualstudio.com/docs/supporting/FAQ#\\_how-to-disable-telemetry-reporting](https://code.visualstudio.com/docs/supporting/FAQ#_how-to-disable-telemetry-reporting)



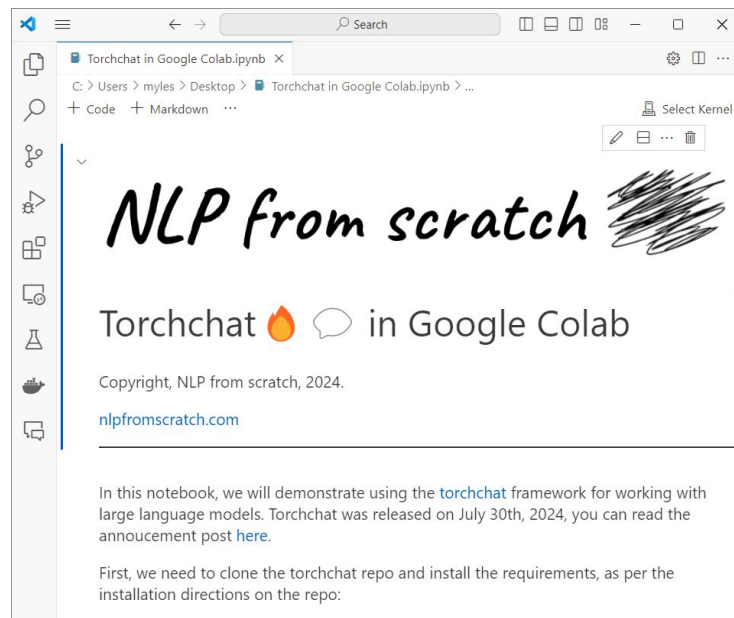


# Jupyter and Markdown in VS Code

- Jupyter files (`.ipynb`) can be natively opened and render automatically in VS Code
- Markdown files (`.ipynb`) can be rendered into a preview using `Cmd/Ctrl + Shift + V`



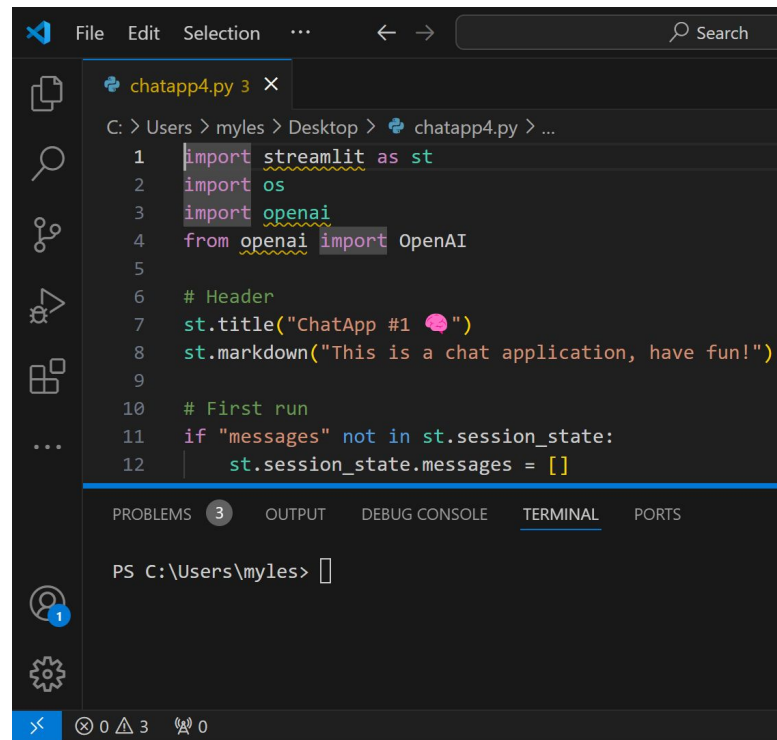
Markdown render in VS Code



Jupyter Notebook in VS Code

# Terminal, Python, etc. in Visual Studio Code

- An integrated terminal can be opened with `~`
- This is useful for debugging and “quick-and-dirty” work without leaving the editor
- There are also useful extensions for Python syntax highlighting, working with CSV
- Git integration is native
- Other extensions for frameworks like Docker, remote file editing, etc.
- Change colour theme with:  
`Cmd/Ctrl + K, Cmd/Ctrl + T`



The screenshot shows the Visual Studio Code interface. The top part is the editor window with a file named 'chatapp4.py'. The code in the editor is as follows:

```
1 import streamlit as st
2 import os
3 import openai
4 from openai import OpenAI
5
6 # Header
7 st.title("ChatApp #1 🗨️")
8 st.markdown("This is a chat application, have fun!")
9
10 # First run
11 if "messages" not in st.session_state:
12     st.session_state.messages = []
```

The bottom part of the screenshot shows the integrated terminal. The terminal prompt is 'PS C:\Users\myles>' and it is currently empty.



# First Steps in NLP (Hands-on)

# End of Part 1

[NLPfor.me](https://nlpfor.me)

PWYC Microcourse in Natural Language Processing  
October 2024

**Part 1 – Course Overview & Introduction to NLP**

**Monday, October 7th, 2024**



[nlpfor.me](https://nlpfor.me)

*NLP from scratch* 