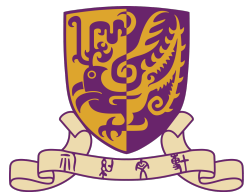
LING3401 Linguistics and Information Technology

Tutorial: Machine learning basics

Yige Chen

The Chinese University of Hong Kong

February 5, 2025



Acknowledgment

- Some of the tutorial materials are based on: Dan Jurafsky and James H. Martin. *Speech and Language Processing* (3rd ed. draft). 2024.
- I owe my gratitude to my former advisor at the University of Washington. The way I deliver materials is heavily influenced by her approach.
- GPT-4o and DeepSeek-R1 helped me write more than half of today's codes. Thanks GPT and DeepSeek!

Programming with LLMs

- What is the easiest way to program (i.e., to write Python codes) these days?
- Short answer: Ask LLMs!
  - Just tell the LLM your need, like what you want your code to do, what are your code's input and output, and any other specific need!
  - If you want an LLM to modify the code, provide the original code and let it know what modification you want
  - If your code (including the code generated by LLMs) does not run or does not work as you expected, just send the code to LLM and tell it that it doesn't work
  - Don't worry at all if you feel that this is not programming – a lot of developers these days do that as well!

Programming with LLMs

- Why LLMs can program?
- Short answer: Codes and scripts in different programming languages are also fed to LLMs as their input during pre-training
- ...and fine-tuning (user's query + codes)

Have a try!

- Let's try to write codes for a rock-paper-scissors game!
- Open your Colab notebook
- If you have access to an LLM, use your own
- If not, I'd recommend Poe – I think you can use some of its available models even without a subscription
  - Alternatively, CUHK students may have access to OpenAI's LLMs upon request
    https://cuhk-edt.knowledgeowl.com/docs/pilot-chatgpt-service-for-teaching-and-learning
- What do you think you will need to inform the model for it to write the code wrt. the aforementioned purpose?
- Once you get its response, copy and paste it to Colab!

Have a try!

- Let's try to modify some codes!
- Assume you already have your rock-paper-scissors game from your LLM
- What if we want to change the probabilities of the computer to $\{rock : 50\%, paper : 25\%, scissors : 25\%\}$?
- What do you think you will need to inform the model for it to modify the code wrt. the aforementioned purpose?
- Once you get its response, copy and paste it to Colab to replace the original cell(s)!
- Try again! Now is it easier to win with papers?

Have a try!

- Congratulations! You are now a programmer!

More about Python and Colab (or notebook)

- A Colab instance has quite a lot of Python packages already installed
  - If a package your LLM suggested is not installed:
    - Ask your LLM for solution, or
    - `!pip install package-name`
- A cell contains one or more than one line of Python codes
- Python codes are run line by line
- To run a cell means that you want to run all lines in the cell
  - Click "run" to run the cell
  - Click again to interrupt the current running cell
    - So if your code stuck in a cell, try to terminate and either run again or ask LLMs
- To use a GPU (T4 available to free tier users), configure through runtime type

More about Python and Colab (or notebook)

- If a line starts with #, this line will not be run
  - # is used for comments in Python. Other programming languages may use a different symbol
- `print()` prints out anything that you plug in and is useful during debugging

n-gram LMs revisited

- Take a look at Part 2 of today's Colab notebook!

Machine learning basics

- Supervised Learning: learning a mapping from inputs (e.g., text) to outputs (e.g., labels) using labeled data
- Unsupervised Learning: finding patterns or structures in data without labeled outputs

Machine learning basics

- Pop quiz! Are these examples of supervised or unsupervised learning?
  - You are given a dataset of movie reviews. Each review is labeled as either "positive" or "negative". Your task is to train a model to predict whether a new review is positive or negative.
  - You have a collection of news articles, but they are not labeled. Your task is to group the articles into different topics based on patterns in the text.

Training, validation, and testing

- Training, validation, and testing
  - Training set: used to train the model by adjusting weights based on the data.
  - Validation set: used to tune hyperparameters and prevent overfitting.
  - Test set: assesses the final performance of the trained model on unseen data.
- Overfitting and generalization
  - Overfitting: the model learns noise in the training data and performs poorly on new data.
  - Generalization: the model performs well on unseen data by capturing the underlying patterns.
- Evaluation metrics
  - Examples: accuracy, precision, recall, F1-score, perplexity, etc.
  - Metrics are chosen based on the task (e.g., classification vs. generation).

Have a try!

- Take a look at Part 3 of today's Colab notebook, and see the examples of supervised learning!
  - It is a text classification task (sentiment analysis)
  - We are using two machine learning algorithms: support vector machines and logistic regression
  - I also included two optional ML tasks using different models, one with BERT classifier (supervised) and the other with k-means (unsupervised).
- Pay attention to
  - What ML algorithms do they use
  - How their data look different
  - The training/test split. Does it use a validation set? Does it employ cross-validation?
  - What kind of results the model gives
  - How the models are evaluated?
  - How are the results different wrt. ML algorithms?

Miscellaneous

- **Please let me know if you think the tutorials have been too hard**
- Please do not hesitate to ask questions
- We enjoy feedback from you, so please let us know if you feel there's anything we could have done better
- It would be great if you'd bring your laptop to the class every week