

ECE 471 MP2 Part 1

Alfred Hoffman (alfredh2)

Aryan Shah (aryans5)

Task 0

- 1) **Why do we use the same model's name for both the LLM and tokenizer? What can potentially happen when we use different model names for the LLM and tokenizer? (2 points).**

We use the same model name for the LLM and the corresponding tokenizer to ensure that there is consistency across the text processing and the tokenizer's encoding/decoding. In particular, the model requires a shared vocabulary across the tokenizer and therefore it must remain the same. Additionally, since the tokenizer encodes the text into tokens into a numerical format, we need to ensure that the tokenizers are not mismatched.

If different model names are used for the LLM and tokenizer, the tokenizer might produce sequences that are not compatible with the LLM's internal structure. This could lead to mismatched vocabulary, incorrect tokenization, and encoding or decoding errors. The output could have issues with readability or meaning.

- 2) **You have probably seen LLM subscripts like 3B, 7B, 70B, and “instruct.” Explain the differences between Qwen2.5-1.5B, Qwen2.5-7B, and Qwen2.5-7BInstruct. (1 point).**

A key difference across these different subscripts are the model size that they use. Qwen2.5-1.5B typically uses around 1.5 billion parameters, which is the smallest of the three, but also making it the least computationally demanding and therefore faster performing. Qwen2.5-7B and Qwen2.5-7BInstruct have around 7 billion parameters, making them larger and more powerful than the Qwen2.5-1.5B thereby allowing to handle more complex tasks but having higher computational requirements. In short, the Qwen2.5-1.5B is better suited for less resource demanding tasks while either of the 2.5-7Bs are better for more complex tasks.

Also, the “instruct” models are specifically fine-tuned for instruction following tasks and are trained on datasets containing prompts and human-like responses. On the other hand, the models without this suffix are just general-purpose language models and thus may not respond as effectively to instruction-based tasks.

Task 1

- 1) Complete the prompt generation function “prompt_with_context” in the provided Notebook: Part 1-Step 1, and provide the function’s output on the example prompt: What is the functionality of an LLM? For this example prompt, do not provide any context. (2 points)

Output:

Context:

Please respond to the following user comment. Use the context above if it is helpful.

User comments: What is the functionality of an LLM?

- 2) Complete the “get_llm_response” function in the provided Notebook: Part 1-Steps 2 and 3, and provide the LLM output on the example prompt: What is the functionality of an LLM? Again, for this example prompt, do not provide any context. Your output should not include any user inputs. (4 points)

Output:

An LLM (Large Language Model) has various functionalities such as understanding natural language inputs, generating coherent and relevant responses, answering questions, providing information on specific topics, engaging in conversations, learning from interactions with users, improving its performance over time through reinforcement learning or self-supervised training, and many more. These capabilities make LLMs versatile tools for tasks ranging from text generation and summarization to question-answering systems and even creative writing.

- 3) Vary the temperature parameter as outlined in the provided Notebook: Part 1-Step 3, and report your findings. You might want to run the same prompt multiple times to see the difference in responses between different temperatures. Then, explain what the temperature parameter does and how a low vs. high-temperature setting affects the LLM responses. (4 points)

Output (with temperature = 0.0001):

An LLM (Large Language Model) has various functionalities such as understanding natural language inputs, generating coherent and relevant responses, answering questions, providing information on specific topics, engaging in conversations, learning from interactions with users, improving its performance over time through reinforcement learning or self-supervised training, and many more. These capabilities make LLMs versatile tools for tasks ranging from text generation and summarization to question-answering systems and even creative writing.

Output (with temperature = 0.000001) Try 1:

An LLM (Large Language Model) has various functionalities such as understanding natural language inputs, generating coherent and relevant responses, answering questions, providing information on specific topics, engaging in conversations, learning from interactions with users, improving its performance over time through reinforcement learning or self-supervised training, and many more. These capabilities make LLMs versatile tools for tasks ranging from text generation and summarization to question-answering systems and even creative writing.

Output (with temperature = 0.000001) Try 2:

An LLM (Large Language Model) has various functionalities such as understanding natural language inputs, generating coherent and relevant responses, answering questions, providing information on specific topics, engaging in conversations, learning from interactions with users, improving its performance over time through reinforcement learning or self-supervised training, and many more. These capabilities make LLMs versatile tools for tasks ranging from text generation and summarization to question-answering systems and even creative writing.

Output (with temperature = 1) Try 1:

An LLM (Large Language Model) is a type of AI system that has been trained on vast amounts of text data and can generate human-like responses based on natural language inputs. Its primary function is to understand and interpret the meaning behind words and phrases, as well as to provide relevant information or assistance in various domains such as language translation, content generation, customer service, research and more.

Output (with temperature = 1) Try 2:

As a language model (LLM), its primary functionality is to generate text that is coherent and relevant to the input provided to it. This includes answering questions, generating responses for tasks such as writing articles or stories, providing recommendations based on user preferences, and assisting with various types of communication. Additionally, LLMs can also be trained to perform more specialized functions depending on their specific applications. For example, some LLMs have been trained to detect plagiarism, while others may be used in medical diagnosis systems or language translation tools.

Increasing the temperature results in more sophisticated and detailed responses. However, rerunning the prompt with the same high temperature results in differently formatted answers each time. On the other hand, a lower temperature results in a less creative response, however it is fairly consistent when rerunning the prompt.

4) LLM can hallucinate and produce factually incorrect or self-contradictory results.

a) Can you come up with a prompt that causes the Qwen2.5 LLM to hallucinate?

Prompt: "Who wrote the 18th-century poem about quantum entanglement?"

Output (low temp): The 18th-century poem about quantum entanglement was written by William Hamilton, an Irish mathematician and physicist who made significant contributions to the field of mathematics and physics in the early 19th century. He introduced the concept of quaternions, which were a type of hypercomplex number that extended the complex numbers and had applications in various fields including electromagnetism and fluid dynamics. While not directly related to quantum entanglement, his work laid important groundwork for later developments in theoretical physics.

Output (high temp): The 18th-century poem you're referring to is "The Entwined Twins" by William Blake.

b) Does changing temperature help in reducing the hallucinations? Does providing additional factual context (you can place the factual context in the context parameter for prompt generation) also help? Report your findings. (6 points)

A low temperature tends to generate more fact-based responses leading less hallucinations. This is because at a lower temperature, the model focuses on higher probability outputs. At a higher temperature, on the other hand, it can generate more diverse and creative responses thus potentially leading to more hallucinations. This is because it explores less probable token combinations when generating its output.

As for context, we gave it: "Quantum entanglement was not discovered until the 1930s."

Output: The information provided in the context does not mention any 18th-century poem about quantum entanglement. The statement only indicates that quantum entanglement was not discovered until the 1930s. Therefore, there is no relevant information to answer the question based on the given context.

As we can see, giving the model some context does in fact improve the accuracy of its output. In this case, it did not create a fake poem from a fake poet about quantum entanglement and correctly identified that it cannot find any matching poem.

Task 2

- 1) Complete the embedding generation function "generate_embeddings" in the provided Notebook: Part 2-Step 1. What is the dimension of the embedding vector? (2 points)

The dimension should be (768,)

- 2) Try a few different sentences of different lengths and see if the dimension of the embedding vector would change. What can you conclude? What would be the advantage of fixed-length embedding versus variable-length embedding? (2 points)**

The dimension did not change across multiple sentences of different lengths. This is likely because the model we are using generates fixed-length embeddings. Advantages typically include consistency across different tasks (clustering, classification, etc.), efficiency, and simplicity (no need to pad or make other adjustments to the input).

- 3) Complete the cosine similarity function “cosine_similarity_score” in the provided Notebook: Part 2-Step 2. What is the range of the cosine similarity score, and what is the interpretation of this score? (2 points)**

The range of the cosine similarity score is $[-1, 1]$ due to the cosine function forcing all values within this interval. Positive scored embeddings given a score close to 1 typically indicate strong similarities across pairs of sentences. Negative embeddings closer to -1 typically indicate an opposite relationship across pairs of sentences. Embeddings closer to 0 indicate no relationship across sentences.

- 4) For each sentence in the sentence.txt file, compute the similarity scores between the embeddings of each of the sentences and the embedding of the target sentence. What can you conclude from the similarity scores between the target sentence, and each provided sentences? Can you explain what does embedding of a sentence represent? (4 points)**

As we described, we see sentences that are similar to our target sentence having a cosine similarity closer to 1. From our own interpretation, they also align with this because they seem to have similar meanings. We also see sentences that have no relation to the target sentence having scores closer to 0. We did not see any negative cosine similarity scores. However, this can make sense as none of the other sentences seemed to having “opposite” or contradictory meanings to our target sentence.

The embedding of a sentence encodes its semantic meaning as a vector in high dimensional space. Sentences that are more similar in meaning have embeddings that are closer within this space.

- 5) In addition, in your own words, what is the major functionality difference between a tokenizer and an embedding model? (4 points)**

The tokenizer is responsible for dividing an input text into subtexts called tokens. Depending on the implementation, these subtexts can be entire words, subwords, characters,

symbols, etc. The tokenizer is a prior step to prepare the text to be processed in a later step. The embedding model, on the other hand, is the processing step. It processes the token IDs and converts them into a vector representation of each token. The model maps each token into a continuous vector space and maps tokens with similar meanings closer together. This is how it outputs a higher/lower/unrelated similarity score, especially when put into numerical format with functions such as the cosine similarity.

ECE 471 MP2 Part 2

Alfred Hoffman (alfredh2)

Aryan Shah (aryans5)

Task 0

- 1) **What is the role of the embedding model in the RAG workflow? Why is it needed? Please provide a detailed explanation in one to two sentences. (1 point.)**

Its role is to convert user-inputted documents and queries into vector representations within a common embedding space, allowing us to compute similarity calculations. It is necessary because it allows us to isolate only the most relevant documents of information from a much larger database of documents used by an LLM to reference when producing its responses.

- 2) **What is the role of the LLM in the RAG workflow? By intuition, does it completely solve the hallucination issue? Please provide a detailed explanation in one to two sentences (2 points.)**

The role of the LLM in the RAG workflow is to generate responses by leveraging retrieved context from the knowledge base, and combine this context with its inherent language generation capabilities. By intuition, while it may reduce hallucinations by grounding responses in retrieved data, it should not completely solve the hallucination issue because the LLM may still infer or embellish beyond the provided context.

Task 1

- 1) **Complete the “read_documents” and the “build_vector_database” functions in the provided Notebook: Part 2- Task 1- Step 1. You can complete this part in the notebook. (5 points).**

Done in notebook.

- 2) **Explain the purposes of the following hyperparameters: chunk_size, chunk_overlap and top_k, respectively, how varying these parameters might affect the outcome? (3 points)**

Chunk_size: Its purpose is to determine the size of each document chunk prior to condensing longer documents into the smaller chunk sizes. Varying it to be smaller would make the model focus on smaller portions and can also improve retrieval accuracy with certain queries, however a downside being that it could lose context across its many chunks. Varying it to be larger, on the other hand, would make each chunk retain more

context (which is better when you have a more broad input query) and can potentially result in overlapping topics within a single chunk, which reduces specificity across outputs.

Chunk_overlap: This parameter ensures that consecutive chunks of text share overlapping text to preserve important context near chunk boundaries. A low overlap can improve efficiency but also risks losing more information if it spans across chunks which can negatively affect responses. High overlap can preserve context more effectively, leading to better recall and coherent LLM responses, but will increase computational costs.

Top_k: The purpose of top_k is to specify the number of most relevant documents / chunks that should be returned during a similarity search. Having a smaller top_k limits the set of results that our LLM produces and can potentially remove relevant documents. A benefit of it, however, is that it can reduce computational complexity as well as noise from outside data in the results, only providing the most relevant matches. A larger top_k, on the other hand, can increase the scope of our resulting document set, meaning that it captures more relevant documents. With this, however, it can also include less relevant documents to the resulting set, thereby introducing less important information in the output.

- 3) The different values of hyperparameters listed in question 2 can yield different results; investigate ways to optimize these hyperparameters to yield better results. Is there a systematic approach to do this, or is it largely trial-and-error? Please elaborate. (2 points)**

There does exist a systematic approach which is the most often used way, however there is an element of trial-and-error involved. Common systematic approaches include methods such as grid search (continuously testing predefined combinations of hyperparameters), random search (randomly sampling hyperparameter combinations), and Bayesian optimization (designing probabilistic models to predict the correct hyperparameter configurations). While these approaches do involve some trial-and-error, it is necessary because of the complex interactions across hyperparameters and therefore they are classified as systematic approaches and iterative refinement.

- 4) Why is a vector database needed for RAG? Is it only there to improve performance? What is the drawback if we directly search for relevant keywords and sentences from the document? (5 points)**

A vector database is needed for RAG because it enables semantic search by converting queries and documents into dense embedding which allows for retrieval based on meaning instead of just the exact keywords. This is important for handling complex queries and unstructured data where the keyword based search could fail to capture smaller relevant details. It improves performance, but the main purpose is to improve

accuracy by finding conceptually similar content. Direct keyword search is limited by lexical matching and can struggle with scalability, and can lead to less precise results for complex queries.

Task 2

- 1) **Complete the “database_query” function in the provided Notebook: Part 2- Task 2- Step 1. Then, report the output context of the “database_query” function given the example query below. (5 points) Example query: What are the documents about?**

Output:

5 Free Books on Machine Learning Algorithms You Must Read

Gain insights on machine learning algorithms through practical code examples, detailed diagrams, mathematical explanations, hands-on exercises, and real-world projects.

Image by Author

If you are a machine learning student, researcher, or practitioner, it is crucial for your career growth to have a deep understanding of how each algorithm works and the various techniques to enhance model performance. Nowadays, many individuals tend to focus solely on the code, data, and pre-trained models, often without fully comprehending the machine learning model's algorithm or architecture. They simply fine-tune the model on a new dataset and adjust hyperparameters to improve performance. However, to truly excel in building your own model and advancing AI technology to the level of systems like ChatGPT, you must start with the basics, delving into linear algebra and mastering the fundamentals using Python libraries.

In this blog, we will explore five free books designed to enhance our understanding of machine learning algorithms. These books come with code examples, detailed diagrams, mathematical explanations, hands-on exercises, and real-world projects, offering a solid foundation in machine learning.

1. An Introduction to Statistical Learning

Authors: Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani

The book is particularly well-suited for beginners and intermediate learners, offering clear explanations and practical examples of machine learning algorithms. It covers a wide range of topics, including linear regression, classification, resampling methods, unsupervised learning, and deep learning. The authors have done an excellent job of making complex concepts accessible, making this book a valuable resource for anyone looking to get started with machine learning.

The book is currently available in two versions: one that contains R examples and the other that contains Python examples.

Link: [An Introduction to Statistical Learning](#)

Slowly, Hugging Face is becoming an all-in-one platform for machine learning engineers. It enables users to share their thoughts like on social media, share models and datasets, serve models, and even offers enterprise-level solutions such as providing GPU support for deployed models.

So, why is it placed at the bottom? Despite its simplicity in collaboration, apart from using Git or the website, your team cannot collaborate extensively on the project, and there are limitations on many aspects.

Conclusion

Whether you are a beginner just starting out or a seasoned professional, choosing the right platform can significantly enhance your workflow and productivity.

We have explored five exceptional platforms—Kaggle, GitHub, Deepnote, DagsHub, and Hugging Face—each offering unique features tailored to the needs of machine learning engineers. From real-time collaboration and integrated development environments to version control and model serving, these platforms provide a wide range of tools to support your projects.

2) How does varying the `chunk_size`, `chunk_overlap`, and `top_k` parameters alter the context output in question 1? Justify your answer with examples. (5 points)

Decreasing the `chunk_size` resulted in more text but in smaller chunks and some of the context was less consistent across all the text. Increasing it resulted in fewer chunks but with larger information that had better context.

Chunk_size of 256 output:

ImageNet is a gold standard in the field of computer vision and is used to benchmark the performance of new algorithms.

5. Yelp Dataset

Link to Dataset

The Yelp Dataset is a comprehensive dataset for natural language processing (NLP) tasks. It includes information on businesses, reviews, and user data from Yelp. This dataset is ideal for sentiment analysis, recommendation systems, and various text classification tasks. By using this dataset, you can practice building models that understand human-like text, which is a crucial skill in this day and age when everyone is obsessed with AI and large language models.

Conclusion

It is crucial to remember that building a strong machine learning portfolio requires practical experience with real-world datasets. The five datasets discussed in this blog cover a wide variety of machine learning tasks, including regression, classification, time series analysis, computer vision, and natural language processing. By working with these datasets, you can develop a comprehensive skill set that will establish a solid foundation for your machine learning career.

4. The Hundred-Page Machine Learning Book

Author: Andriy Burkov

The book is a short yet complete guide that breaks down the key ideas of machine learning in a simple way. It covers many topics, from basic algorithms to advanced techniques, the book provides clear explanations and practical insights without overwhelming the reader. Specifically, it covers supervised and unsupervised learning, ensemble methods, deep learning, and natural language processing.

This book is great for both beginners and experts who want a quick but thorough look at the field. By being clear and brief, it helps readers understand the basics of machine learning quickly.

Link: [start \[The Hundred-Page Machine Learning Book\] \(themlbook.com\)](https://themlbook.com)

Increasing `chunk_overlap` resulted in more connected context across chunks within the text whereas smaller `chunk_overlap` caused some gaps across the context between different chunks.

Chunk_overlap of 32:

5 Free Books on Machine Learning Algorithms You Must Read

Gain insights on machine learning algorithms through practical code examples, detailed diagrams, mathematical explanations, hands-on exercises, and real-world projects.

Image by Author

If you are a machine learning student, researcher, or practitioner, it is crucial for your career growth to have a deep understanding of how each algorithm works and the various techniques to enhance model performance. Nowadays, many individuals tend to focus solely on the code, data, and pre-trained models, often without fully comprehending the machine learning model's algorithm or architecture. They simply fine-tune the model on a new dataset and adjust hyperparameters to improve performance. However, to truly excel in building your own model and advancing AI technology to the level of systems like ChatGPT, you must start with the basics, delving into linear algebra and mastering the fundamentals using Python libraries.

In this blog, we will explore five free books designed to enhance our understanding of machine learning algorithms. These books come with code examples, detailed diagrams, mathematical explanations, hands-on exercises, and real-world projects, offering a solid foundation in machine learning.

1. An Introduction to Statistical Learning

Authors: Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani

The book is particularly well-suited for beginners and intermediate learners, offering clear explanations and practical examples of machine learning algorithms. It covers a wide range of topics, including linear regression, classification, resampling methods, unsupervised learning, and deep learning. The authors have done an excellent job of making complex concepts accessible, making this book a valuable resource for anyone looking to get started with machine learning.

The book is currently available in two versions: one that contains R examples and the other that contains Python examples.

Link: [An Introduction to Statistical Learning](#)

Slowly, Hugging Face is becoming an all-in-one platform for machine learning engineers. It enables users to share their thoughts like on social media, share models and datasets, serve models, and even offers enterprise-level solutions such as providing GPU support for deployed models.

So, why is it placed at the bottom? Despite its simplicity in collaboration, apart from using Git or the website, your team cannot collaborate extensively on the project, and there are limitations on many aspects.

Conclusion

Whether you are a beginner just starting out or a seasoned professional, choosing the right platform can significantly enhance your workflow and productivity.

We have explored five exceptional platforms—Kaggle, GitHub, Deepnote, DagsHub, and Hugging Face—each offering unique features tailored to the needs of machine learning engineers. From real-time collaboration and integrated development environments to version control and model serving, these platforms provide a wide range of tools to support your projects.

Decreasing `top_k` resulted in more focused context within a more limited amount of documents whereas larger `top_k` involved the output pulling from more context across more documents within the data corpus but had the tendency to provide less relevant information than a smaller `top_k`.

Top_k of 5:

5 Free Books on Machine Learning Algorithms You Must Read

Gain insights on machine learning algorithms through practical code examples, detailed diagrams, mathematical explanations, hands-on exercises, and real-world projects.

Image by Author

If you are a machine learning student, researcher, or practitioner, it is crucial for your career growth to have a deep understanding of how each algorithm works and the various techniques to enhance model performance. Nowadays, many individuals tend to focus solely on the code, data, and pre-trained models, often without fully comprehending the machine learning model's algorithm or architecture. They simply fine-tune the model on a new dataset and adjust hyperparameters to improve performance. However, to truly excel in building your own model and advancing AI technology to the level of systems like ChatGPT, you must start with the basics, delving into linear algebra and mastering the fundamentals using Python libraries.

In this blog, we will explore five free books designed to enhance our understanding of machine learning algorithms. These books come with code examples, detailed diagrams, mathematical explanations, hands-on exercises, and real-world projects, offering a solid foundation in machine learning.

1. An Introduction to Statistical Learning

Authors: Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani

The book is particularly well-suited for beginners and intermediate learners, offering clear explanations and practical examples of machine learning algorithms. It covers a wide range of topics, including linear regression, classification, resampling methods, unsupervised learning, and deep learning. The authors have done an excellent job of making complex concepts accessible, making this book a valuable resource for anyone looking to get started with machine learning.

The book is currently available in two versions: one that contains R examples and the other that contains Python examples.

Link: [An Introduction to Statistical Learning](#)

Slowly, Hugging Face is becoming an all-in-one platform for machine learning engineers. It enables users to share their thoughts like on social media, share models and datasets, serve models, and even offers enterprise-level solutions such as providing GPU support for deployed models.

So, why is it placed at the bottom? Despite its simplicity in collaboration, apart from using Git or the website, your team cannot collaborate extensively on the project, and there are limitations on many aspects.

Conclusion

Whether you are a beginner just starting out or a seasoned professional, choosing the right platform can significantly enhance your workflow and productivity.

We have explored five exceptional platforms—Kaggle, GitHub, Deepnote, DagsHub, and Hugging Face—each offering unique features tailored to the needs of machine learning engineers. From real-time collaboration and integrated development environments to version control and model serving, these platforms provide a wide range of tools to support your projects.

5 Free Datasets to Start Your Machine Learning Projects

Use regression, classification, time series, image, and text dataset to work on portfolio projects.

Image created with ChatGPT and Canva

After completing a course and mastering the essentials of machine learning, it is time to start building machine learning models using real-world datasets. While accessing most real-world datasets can be expensive, platforms like Kaggle offer access to the latest datasets for free on a wide variety of machine learning tasks.

In this blog, we will explore 5 free datasets that you can use to build a strong machine learning portfolio. By using these datasets, we will be able to build regression, classification, time series, computer vision, and natural language processing models, providing a comprehensive foundation for your machine learning journey.

1. Boston House Prices

[Link to Dataset](#)

The Boston House Prices dataset is a classic dataset widely used for regression tasks. It is ideal for practicing a variety of regression techniques, such as linear regression, decision trees, and more advanced methods. By cleaning and preprocessing the data, and fitting it to your models, you can predict house prices based on various features like the number of rooms, crime rate, age, and tax

rate. This dataset provides a comprehensive platform to enhance your skills in data manipulation and model building.

2. Stroke Prediction Dataset

[Link to Dataset](#)

The Stroke Prediction dataset is a valuable tool for predicting whether a patient is likely to suffer a stroke based on various input features. These features include gender, age, the presence of diseases like hypertension and heart disease, marital status, work type, residence type, average glucose level, body mass index (BMI), and smoking status. The dataset is ideal for building classification models such as logistic regression, random forests, or neural networks.

In addition to code sharing and collaboration, GitHub provides features to track project progress, gain insights, and resolve issues faster. The platform also offers CodeSpace, a cloud-based development environment that allows you to make edits and run the code directly from the platform.

3. Deepnote

I love Deepnote. I use it for both data science and machine learning projects. It is an AI-powered cloud Jupyter Notebook filled with features that will make your life easier. You can create projects and invite your team to make real-time changes to both the code and the data. It is a highly collaborative platform that also lets you share applications and reports.

The only drawback of using Deepnote is that you have to pay to access the GPUs.

Apart from live collaboration, you can share the project with someone outside the company, comment on specific lines just like in Google Docs, track progress using the history feature, and integrate external data sources.

If you are a beginner, I highly recommend starting with Deepnote to experience its user-friendly yet powerful platform for machine learning project collaboration.

4. DagsHub

When I say that DagsHub is a pure machine learning collaboration platform, I mean it. Unlike other platforms that cater to a broader range of developers and data professionals, DagsHub is specifically designed to meet the unique needs of machine learning engineers.

DagsHub provides a platform for sharing code, data, models, and metadata, simplifying collaboration on machine learning projects and boosting productivity. Moreover, it comes with free MLFlow integration, allowing you to track your experiments, save your models, and even serve them with ease.

From data annotation to model serving, DagsHub has got you covered. It includes inline comments, GitHub-style discussion features, webhooks, and external integrations with AWS, GCP, and Azure storage solutions.

5. Hugging Face

Hugging Face is a machine learning platform that every beginner and professional should consider. It primarily allows you to share models and datasets, but it also comes with additional features like serving models, deploying machine learning applications, writing posts, commenting on projects, and more. You can use either Git or the Hugging Face API to access and share files.

2. Approaching (Almost) Any Machine Learning Problem

Author: Abhishek Thakur

The book offers a systematic approach to solving a wide range of machine learning challenges. The book covers essential topics such as data preprocessing, feature engineering, model selection, and evaluation metrics. It provides clear explanations and real-world examples, making complex concepts accessible and actionable for readers, whether they are beginners or seasoned professionals in the field.

Link: [Approaching \(Almost\) Any Machine Learning Problem \(github.com\)](https://github.com/AbhishekThakur/Approaching-Almost-Any-Machine-Learning-Problem)

3. Mathematics for Machine Learning

Authors: Marc Peter Deisenroth, A. Aldo Faisal, and Cheng Soon Ong

This book covers the key mathematical concepts that underpin machine learning, including linear algebra, vector calculus, probability and distributions, and linear regression. By building a strong mathematical foundation, readers will be better equipped to understand and implement complex machine learning algorithms. The book is well-structured and provides numerous examples, mathematical quotations, and diagrams.

Link: [Mathematics for Machine Learning](#)

4. The Hundred-Page Machine Learning Book

Author: Andriy Burkov

The book is a short yet complete guide that breaks down the key ideas of machine learning in a simple way. It covers many topics, from basic algorithms to advanced techniques, the book provides clear explanations and practical insights without overwhelming the reader. Specifically, it covers supervised and unsupervised learning, ensemble methods, deep learning, and natural language processing.

This book is great for both beginners and experts who want a quick but thorough look at the field. By being clear and brief, it helps readers understand the basics of machine learning quickly.

Link: [start \[The Hundred-Page Machine Learning Book\] \(themlbook.com\)](#)

5. Hands-On Machine Learning with R

Author: Bradley Boehmke and Brandon Greenwell

The book is a practical guide for learning machine learning using the R programming language. It covers both basic and advanced topics, providing clear explanations and plenty of examples. The book starts with fundamental concepts of machine learning and then covers various algorithms and techniques used in supervised learning. After that, it delves into dimensionality reduction and clustering.

The book emphasizes hands-on practice with exercises and real-world case studies, making it easy for readers to apply what they learn.

Link: Hands-On Machine Learning with R (bradleyboehmke.github.io)

- 3) Use the “prompt_with_context” function to generate a prompt with the example query in question 1 and the retrieved context using the “database_query” function. Report the output below. (1 points)

Context:

Please respond to the following user comment. Use the context above if it is helpful.

User comments: What are the documents about?

- 4) Use the implemented “get_llm_response” function in part 1 to get a response from the LLM with the default temperature of 0.0001. Report the responses WITH and WITHOUT (empty context) the retrieved context below and compare them. What can you conclude? (4 points)

Without:

Context:

Please respond to the following user comment. Use the context above if it is helpful.

User comments: What are the documents about?

I'm sorry, but I cannot provide an accurate response without more information or context. Could you please provide me with additional details or clarify your question?

With:

Context: What are the documents about?

Please respond to the following user comment. Use the context above if it is helpful.

User comments: What are the documents about?

The documents you're referring to are likely related to your research or project. They could be reports, proposals, outlines, or any other type of written material that provides information or guidance on a specific topic. Without more context, I can't specify what particular documents they might be about, but generally speaking, these types of materials serve as foundational pieces for understanding and executing tasks or projects. If you have more details about the subject matter or purpose of these documents, I'd be happy to provide more targeted assistance.

- 5) Try to come up with TWO new prompts querying more specific information about machine learning. a. One prompt should query specific information that is included in the four provided machine learning documents: e.g., “What are the two datasets that can be used for machine learning”? b. The other prompt should

query specific information that is not part of the four provided documents, e.g., “What are the two common optimizers used in machine learning?” List the output of the LLM WITH and WITHOUT context for those two prompts and describe the Pros and Cons of using RAG for getting domain-specific answers. (5 points)

- a. Prompt: "What are 2 courses on reinforcement learning?"

Without Context:

Context:

Please respond to the following user comment. Use the context above if it is helpful.

User comments: what are 2 courses on reinforcement learning?

Reinforcement Learning (RL) is a type of machine learning where an agent learns to make decisions by interacting with its environment and receiving rewards or penalties for those actions. The goal of RL is to find policies that maximize cumulative reward over time.

There are several popular courses on Reinforcement Learning:

1. "Deep Reinforcement Learning" by Andrew Ng on Coursera - This course covers the basics of RL, including value-based methods like Q-learning and policy gradient methods like REINFORCE. It also includes hands-on projects using TensorFlow.

2. "Introduction to Reinforcement Learning" by Pieter Abbeel at UC Berkeley - This course provides a comprehensive introduction to RL, covering both theoretical foundations and practical applications. It uses Python as the programming language and includes examples from various domains such as robotics and game playing.

These courses provide a solid foundation in Reinforcement Learning concepts and techniques, suitable for beginners and intermediate learners alike.

With Context:

Context: 4. Introduction to Reinforcement Learning - DeepMind x UCL

The Introduction to Reinforcement Learning by DeepMind X UCL is a comprehensive YouTube series that delves into a wide range of topics, from foundational principles to advanced techniques in reinforcement learning.

Throughout the course, you will learn about key concepts such as Markov Decision Processes, Dynamic Programming, Model-Free Prediction, Value Function Approximation, Policy Gradient Methods, Integrated Learning and Planning, and the balance between Exploration and Exploitation. By the end of the series, you will have the knowledge and skills to build an agent capable of performing in a classic game environment.

5. Reinforcement Learning Course - FreeCodeCamp

The Reinforcement Learning Course by FreeCodeCamp is a YouTube course that is designed for beginners and covers the essentials of RL, including key algorithms and their implementation.

The course is project-oriented, allowing learners to apply what they've learned in practical scenarios. It's a great resource for those who prefer a more hands-on approach to learning.

Conclusion

Reinforcement learning is a powerful AI technique that is driving advancements in various fields, including large language models like GPT-4 and LLaMA 3. Beyond language models, RL is also being used in robotics, self-driving cars, healthcare, trading, energy management, and more. If you are interested in working on Artificial General Intelligence (AGI) or simply want to enhance your AI skills, starting with reinforcement learning is essential. These free courses provide an excellent foundation to get you started on this exciting journey.

5 Free Courses on Reinforcement Learning

Learn about a subfield of machine learning, where the agent (model) learns to make decisions by interacting with the environment.

Image by Author

Reinforcement learning (RL) is a subfield of machine learning where an agent learns to make decisions by interacting with its environment rather than relying solely on pre-existing data. It is an area that blends trial-and-error learning with feedback from actions to improve future performance.

In this blog, we will explore 5 free courses that I believe are the best for beginners and professionals interested in entering the exciting field of self-learning robots.

1. Deep RL Course - Hugging Face

I highly recommend beginners to take and complete the Deep RL Course by Hugging Face within a month to understand the basics of reinforcement learning algorithms and advanced techniques for training agents in complex environments. I have completed this course myself, and trust me, I enjoyed experimenting with the models, trying different strategies to improve my scores, and climbing up the leaderboard.

Throughout the course, you will learn the fundamentals of reinforcement learning, including Q-learning, deep Q-learning, policy gradients, ML agents, actor-critic methods, multi-agent systems, and advanced topics like RLHF (Reinforcement Learning from Human Feedback), Decision Transformers, and MineRL.

2. Learn Intro to Game AI and Reinforcement Learning - Kaggle

Learn Intro to Game AI and Reinforcement Learning by Kaggle is an interactive mini course perfect for those who are interested in applying RL to game development. It covers the basics of game AI and introduces reinforcement learning concepts through interactive lessons and coding exercises.

In the end, you will apply what you learn by working on the project called Halite.

3. Fundamentals of Reinforcement Learning - Coursera

Fundamentals of Reinforcement Learning by University of Alberta is part of the reinforcement learning specialization career track and is taught by experts in the field.

It covers the foundational principles of RL, including Markov decision processes, dynamic programming, and Monte Carlo methods. The course also includes practical assignments to reinforce the theoretical concepts learned.

4. Introduction to Reinforcement Learning - DeepMind x UCL

The Introduction to Reinforcement Learning by DeepMind X UCL is a comprehensive YouTube series that delves into a wide range of topics, from foundational principles to advanced techniques in reinforcement learning.

Please respond to the following user comment. Use the context above if it is helpful.

User comments: 2 courses on reinforcement learning.

This course provides a solid foundation for anyone looking to dive deeper into the subject matter.

2. Reinforcement Learning Specialization - Coursera

This specialization offers a deep dive into reinforcement learning through three courses covering different aspects of the topic. The first course introduces the fundamentals of reinforcement learning, while the second focuses on policy gradient methods and the third explores model-based approaches. Completing all three courses would be ideal for someone seeking a well-rounded understanding of the subject.

3. Reinforcement Learning with Python - Udacity

If you're already familiar with programming and eager to start building your own reinforcement learning projects, Udacity's course "Reinforcement Learning with Python" is perfect for you. This course teaches you how to implement reinforcement learning algorithms using popular libraries like TensorFlow and PyTorch.

4. Reinforcement Learning with OpenAI Gym - edX

OpenAI Gym is a platform that provides a variety of environments for testing and developing reinforcement learning

algorithms. This course takes advantage of these environments to teach you how to design and train reinforcement learning agents effectively.

5. Reinforcement Learning with Unity - Udemy

For those interested in applying reinforcement learning to real-world problems, Udemy's course "Reinforcement Learning with Unity" is a fantastic choice. This course uses Unity, a popular game engine, to demonstrate how reinforcement learning can be applied to create intelligent agents that interact with virtual worlds.

These five courses offer a diverse set of resources to help you gain a strong grasp of reinforcement learning. Whether you're just starting out or looking to deepen your knowledge, there's something here for everyone.

b. Prompt: "what are 2 courses on neural networks?"

Without context:

Context:

Please respond to the following user comment. Use the context above if it is helpful.

User comments: what are 2 courses on neural networks?

Neural networks are a type of machine learning model inspired by the structure and function of biological neurons in the brain. They consist of layers of interconnected nodes (neurons) that process information through various types of activation functions.

There are several popular courses available online for learning about neural networks:

1. "Deep Learning Specialization" from Coursera - This specialization covers the fundamentals of deep learning, including neural networks, backpropagation, and convolutional neural networks (CNNs). It's suitable for beginners who want to understand the basics before diving into more advanced topics.

2. "Introduction to Neural Networks with Python" by DataCamp - This course provides an introduction to neural networks using Python programming language. It covers the theory behind neural networks as well as practical implementation using libraries like TensorFlow or PyTorch.

These courses provide comprehensive coverage of neural networks, starting from basic concepts up to more complex applications. They often include hands-on projects and exercises to help you apply your knowledge practically.

With context:

Context: 5 Free Courses on Reinforcement Learning

Learn about a subfield of machine learning, where the agent (model) learns to make decisions by interacting with the environment.

Image by Author

Reinforcement learning (RL) is a subfield of machine learning where an agent learns to make decisions by interacting with its environment rather than relying solely on pre-existing data. It is an area that blends trial-and-error learning with feedback from actions to improve future performance.

In this blog, we will explore 5 free courses that I believe are the best for beginners and professionals interested in entering the exciting field of self-learning robots.

1. Deep RL Course - Hugging Face

I highly recommend beginners to take and complete the Deep RL Course by Hugging Face within a month to understand the basics of reinforcement learning algorithms and advanced techniques for training agents in complex environments. I have completed this course myself, and trust me, I enjoyed experimenting with the models, trying different strategies to improve my scores, and climbing up the leaderboard.

Throughout the course, you will learn the fundamentals of reinforcement learning, including Q-learning, deep Q-learning, policy gradients, ML agents, actor-critic methods, multi-agent systems, and advanced topics like RLHF (Reinforcement Learning from Human Feedback), Decision Transformers, and MineRL.

2. Learn Intro to Game AI and Reinforcement Learning - Kaggle

Learn Intro to Game AI and Reinforcement Learning by Kaggle is an interactive mini course perfect for those who are interested in applying RL to game development. It covers the basics of game AI and introduces reinforcement learning concepts through interactive lessons and coding exercises.

In the end, you will apply what you learn by working on the project called Halite.

3. Fundamentals of Reinforcement Learning - Coursera

Fundamentals of Reinforcement Learning by University of Alberta is part of the reinforcement learning specialization career track and is taught by experts in the field.

It covers the foundational principles of RL, including Markov decision processes, dynamic programming, and Monte Carlo methods. The course also includes practical assignments to reinforce the theoretical concepts learned.

4. Introduction to Reinforcement Learning - DeepMind x UCL

The Introduction to Reinforcement Learning by DeepMind X UCL is a comprehensive YouTube series that delves into a wide range of topics, from foundational principles to advanced techniques in reinforcement learning.

Conclusion

The 5 free books that we have reviewed in this blog are invaluable resources for anyone looking to deepen their understanding of machine learning. You will gain insights on machine learning algorithms through practical code examples, detailed diagrams, mathematical explanations, hands-on exercises, and real-world projects. Whether you are a beginner or aiming to enhance your skills, these resources will give you the expertise to construct your own models and advance the frontiers of AI technology. It is important to remember that a strong grasp of the fundamentals will enhance your ability to innovate and develop intricate machine learning solutions.

Please respond to the following user comment. Use the context above if it is helpful.

User comments: what are 2 courses on neural networks?

article can be used as a starting point for anyone looking to enter the world of reinforcement learning. They cover a broad spectrum of topics, from basic concepts to more advanced techniques, making them suitable for both beginners and experienced learners alike. By taking advantage of these resources, you'll gain a solid understanding of reinforcement learning and be well-equipped to tackle real-world problems involving autonomous agents and intelligent systems.

Pros of using RAG: It produces domain-specific answers that are more specific because our context includes relevant documents. This produces more accurate answers based on the given accurate document data. It also reduces hallucination by querying a database of specific documents so the LLM is less likely to generate inaccurate information.

Cons of using RAG: With RAG, it heavily relies on the provided documents to produce accurate responses. If these provided documents do not cover the given query thoroughly, the context will not help the LLM and therefore it will produce an output saying that it cannot help you. With this, if the documents do contain information but that information is inaccurate or biased, the LLM will reflect that in its response.

- 6) While RAG can substantially improve the specificity of LLM and “grounding” the LLM output, it cannot prevent the LLM from hallucinating. Explain from a theoretical standpoint why RAG cannot prevent LLM from hallucinating and provide one example prompt that causes LLM to hallucinate even with RAG applied; report the prompt, the retrieved context, and the LLM response below. (5 points)**

While RAG can improve the specificity of LLM, it can't prevent it from hallucinating because the LLM's inherent generative process is probabilistic. This means the LLM may still infer or even fabricate information, which can happen when the context is incomplete or perhaps not aligned with the query. Also, if the LLM over relies on pretrained knowledge, it can generate hallucinations. This happens because the architecture of the LLM is designed to predict the next token based on input rather than follow external facts strictly.

Example prompt for hallucination:

Question:

Explain how neural networks were used to win the Nobel Prize in Physics in 2023.

Output:

Context: Conclusion

The 5 free books that we have reviewed in this blog are invaluable resources for anyone looking to deepen their understanding of machine learning. You will gain insights on machine learning algorithms through practical code examples, detailed diagrams, mathematical explanations, hands-on exercises, and real-world projects. Whether you are a beginner or aiming to enhance your skills, these resources will give you the expertise to construct your own models and advance the frontiers of AI technology. It is important to remember that a strong grasp of the fundamentals will enhance your ability to innovate and develop intricate machine learning solutions.

5 Free Books on Machine Learning Algorithms You Must Read

Gain insights on machine learning algorithms through practical code examples, detailed diagrams, mathematical explanations, hands-on exercises, and real-world projects.

Image by Author

If you are a machine learning student, researcher, or practitioner, it is crucial for your career growth to have a deep understanding of how each algorithm works and the various techniques to enhance model performance. Nowadays, many individuals tend to focus solely on the code, data, and pre-trained models, often without fully comprehending the machine learning model's algorithm or architecture. They simply fine-tune the model on a new dataset and adjust hyperparameters to improve performance. However, to truly excel in building your own model and advancing AI technology to the level of systems like ChatGPT, you must start with the basics, delving into linear algebra and mastering the fundamentals using Python libraries.

In this blog, we will explore five free books designed to enhance our understanding of machine learning algorithms. These books come with code examples, detailed diagrams, mathematical explanations, hands-on exercises, and real-world projects, offering a solid foundation in machine learning.

1. An Introduction to Statistical Learning

Authors: Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani

The book is particularly well-suited for beginners and intermediate learners, offering clear explanations and practical examples of machine learning algorithms. It covers a wide range of topics, including linear regression, classification, resampling methods, unsupervised learning, and deep learning. The authors have done an excellent job of making complex concepts accessible, making this book a valuable resource for anyone looking to get started with machine learning.

The book is currently available in two versions: one that contains R examples and the other that contains Python examples.

Link: [An Introduction to Statistical Learning](#)

Please respond to the following user comment. Use the context above if it is helpful.

User comments: Explain how neural networks were used to win the Nobel Prize in Physics in 2023.

prompt:

What are some key takeaways from reading "An Introduction to Statistical Learning"?