Large Language Models ∷ CHEAT SHEET

Large Language Models (LLMs)

LLMs are artificial intelligence models that can generate human-like text, based on patterns found in massive amounts of training data. They are used in applications such as language translation, chatbots, and content creation.

Some popular LLMs

Some popular LLMs include GPT-3 (Generative Pretrained Transformer by OpenAI, BERT (Bidirectional Encoder Representations from Transformers) by Google, and XLNet (eXtreme MultiLingual Language Model) by Carnegie Mellon University and Google.

BigScience	BLOOM	176B	July 2022
	T0pp	11B	October 2021
EleutherAI	GPT-J	6B	July 2021
	GPT-NeoX	20B	February 2022
清华大学 Tsinghua University	GLM	130B	August 2022
Google Research	UL2	20B	October 2022
	T5	11B	February 2020
Meta AI	OPT	175B	June 2022
	OPT	66B	June 2022
Yandex	YaLM	100B	June 2022

Choose between LLMs
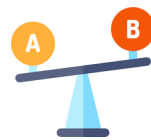
When comparing different models, it's important to consider their architecture, the size of the model, the amount of training data used, and their performance on specific NLP tasks.

Components of LLMs

LLMs typically consist of an encoder, a decoder, and attention mechanisms. The encoder takes in input text and converts it into a set of hidden representations, while the decoder generates the output text. The attention mechanisms help the model focus on the most relevant parts of the input text.

Applications of LLMs

- LLMs are used in a wide range of applications, **including language translation, chatbots, content creation, and text summarization**.

- They can also be used to improve search engines, voice assistants, and virtual assistants.

How are LLMs trained?

LLMs are trained using a process called unsupervised learning. This involves feeding the model massive amounts of text data, such as books, articles, and websites, and having the model learn the patterns and relationships between words and phrases in the text. The model is then fine-tuned on a specific task, such as language translation or text summarization.

Preprocessing

Text normalization is the process of converting text to a standard format, such as lowercasing all text, removing special characters, and converting numbers to their written form.
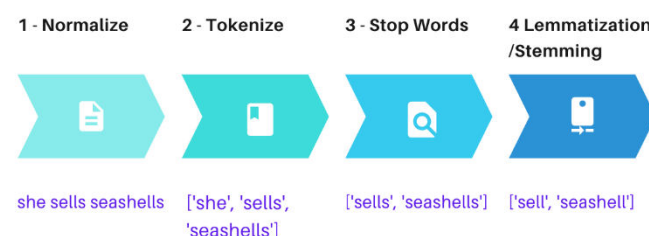
Tokenization is the process of breaking down text into individual units, such as words or phrases. This is an important step in preparing text data for NLP tasks.

Stop Words are common words that are usually removed during text processing, as they do not carry much meaning and can introduce noise or affect the results of NLP tasks. Examples of stop words include "the," "a," "an," "in," and "is."

Lemmatization is the process of reducing words to their base or dictionary form, by taking into account their part of speech and context. It is a more sophisticated technique than stemming and produces more accurate results, but it is computationally more expensive.

Stemming and lemmatization are techniques used to reduce words to their base form. This helps to reduce the dimensionality of the data and improve the performance of models.

Example Text: she sells seashells

1 - Normalize	2 - Tokenize	3 - Stop Words	4 Lemmatization /Stemming
she sells seashells	['she', 'sells', 'seashells']	['sells', 'seashells']	['sell', 'seashell']

Fine-Tuning

Fine-tuning is the process of training a pre-trained large language model on a specific task using a smaller dataset. This allows the model to learn task-specific features and improve its performance.

The fine-tuning process typically involves freezing the weights of the pre-trained model and only training the task-specific layers.

When fine-tuning a model, it's important to consider factors such as the size of the fine-tuning dataset, the choice of optimizer and learning rate, and the choice of evaluation metrics

Example of fine-tuning LLMs

- Model Cost: $500 - $5000 per month, depending on the size and complexity of the language model
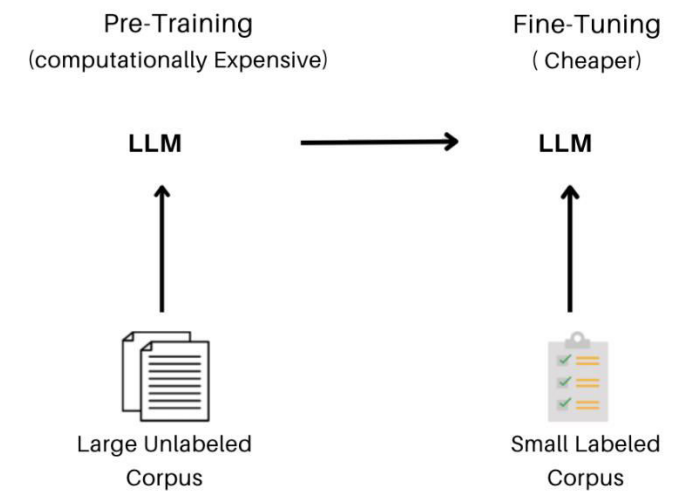
- GPU size: NVIDIA GeForce RTX 3080 or higher

- Number of GPUs: 1-4, depending on the size of the language model and the desired speed of fine-tuning. For example, fine-tuning the GPT-3 model, which is one of the largest language models available, would require a minimum of 4 GPUs.

- The size of the data that GPT-3 is fine-tuned on can vary greatly depending on the specific use case and the size of the model itself. GPT-3 is one of the largest language models available, with over 175 billion parameters, so it typically requires a large amount of data for fine-tuning to see a noticeable improvement in performance.

Note: fine-tuning GPT-3 on a small dataset of only a few gigabytes may not result in a significant improvement in performance, while fine-tuning on a much larger dataset of several terabytes could result in a substantial improvement. The size of the fine-tuning data will also depend on the specific NLP task the model is being fine-tuned for and the desired level of accuracy.

This is just one example, and actual costs and GPU specifications may vary depending on the language model, the data it is being fine-tuned on, and other factors. It's always best to check with the language model provider for the latest information and specific recommendations for fine-tuning.

Pre-Training (computationally Expensive) → Fine-Tuning (Cheaper)

LLM → LLM

Large Unlabeled Corpus → Small Labeled Corpus

Input Representations:

- **Word embeddings:** Each token is replaced by a vector that represents its meaning in a continuous vector space. Common methods for word embeddings include Word2Vec, GloVe, and fastText.

- **Subword embeddings:** Each token is broken down into smaller subword units (e.g., characters or character n-grams), and each subword is replaced by a vector that represents its meaning. This approach can handle out-of-vocabulary (OOV) words and can improve the model's ability to capture morphological and semantic similarities. Common methods for subword embeddings include Byte Pair Encoding (BPE), Unigram Language Model (ULM), and SentencePiece.

- **Positional encodings:** Since LLMs operate on sequences of tokens, they need a way to encode the position of each token in the sequence. Positional encodings are vectors that are added to the word or subword embeddings to provide information about the position of each token.

- **Segment embeddings:** In some LLMs, such as the Transformer, the input sequence can be divided into multiple segments (e.g., sentences or paragraphs). Segment embeddings are added to the word or subword embeddings to indicate which segment each token belongs to.

Ashish Patel • Principal Research Scientist • ashishpatel.ce.2011@gmail.com   Abonia Sojasingarayar • Machine Learning Scientist • aboniaa@gmail.com   Updated: 2023-02

Large Language Models : : CHEAT SHEET

Attention Mechanisms

Self-Attention:

•A mechanism that allows a sequence to weigh the importance of all other elements in the sequence when computing a representation for each element.

•Can capture relationships between different elements in the sequence, making it well-suited for tasks that require modeling long-range dependencies.

•Popularized by the Transformer model.

Multi-Head Attention:

•A variation of self-attention where the attention mechanism is applied multiple times in parallel with different sets of weights.

•Allows the model to attend to different aspects of the input sequence, improving its ability to capture complex patterns and dependencies.

•Each "head" produces a separate output that is concatenated and linearly transformed to produce the final output.

•Also commonly used in the Transformer model.

Multi-Head Attention

Scaled Dot-Product Attention
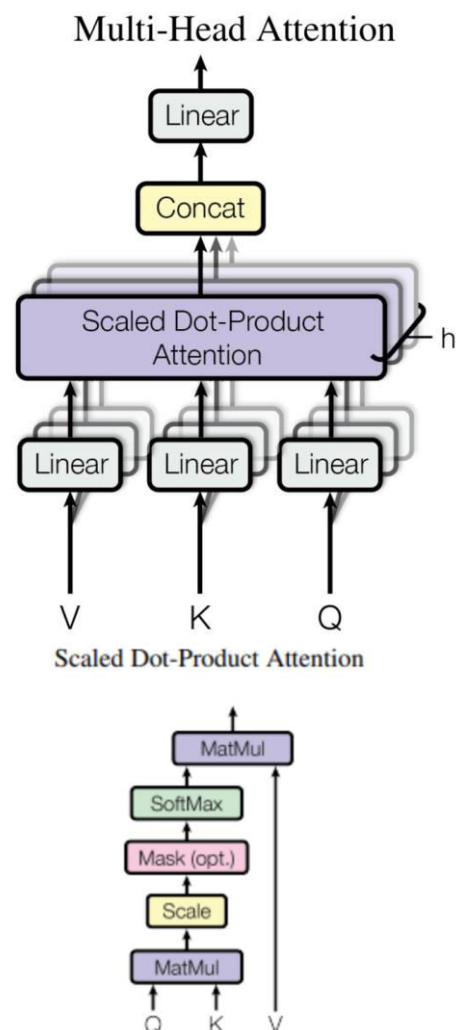
Evaluating LLMs

•**Accuracy** measures the proportion of correctly classified instances out of all instances. This metric is commonly used for text classification tasks such as sentiment analysis, where the goal is to correctly classify a text as positive, negative, or neutral.

•**F1-score** is a metric that takes into account both precision and recall. Precision is the proportion of true positive results out of all predicted positive results, while recall is the proportion of true positive results out of all actual positive results. The F1-score is the harmonic mean of precision and recall, and it provides a balanced measure of model performance on text classification, question answering, and other tasks.

•**Perplexity** : It measures how well a language model predicts the next word in a sequence. The lower the perplexity, the better the model is at predicting the next word. Perplexity is calculated as 2 to the power of the cross-entropy, which is a measure of how well the model's predicted probabilities match the true probabilities of the next word in the sequence.Also text can be evaluated using metrics such as BLEU or ROUGE scores.

Example - ChatGPT

ChatGPT is a large language model developed by OpenAI that has been trained on a diverse range of internet text, including articles, forums, and social media. This training allows ChatGPT to generate human-like text and answer a wide variety of questions on many topics.

Use Cases ChatGPT

Question Answering: ChatGPT can answer factual questions based on the information it has been trained on. Example:

Human : What is the capital of France?
ChatGPT: The capital of France is Paris.

Conversational: ChatGPT can engage in a conversation with a user. Example:

Human : Hi, how are you today?
ChatGPT: Hello! I'm just an AI, so I don't have emotions, but I'm functioning well today. How can I assist you?

Text Generation: ChatGPT can generate text based on a prompt or seed text. Example:

Human : Write a short story about a magical world.
ChatGPT: Once upon a time, in a land far away, there was a magical world filled with wonder and enchantment. The sky was always a brilliant shade of purple, and the stars twinkled like diamonds in the sky. The people of this world possessed magical powers that they used to create incredible wonders and to help those in need. One day, a young woman discovered that she too had magical powers, and set out on an incredible journey to explore the wonders of this magical world and to hone her skills.

Summarization: ChatGPT can summarize a long text or article. Example:

Human : Please summarize this article about the benefits of exercise.
ChatGPT: The article discusses the various physical and mental health benefits of exercise, including improved cardiovascular health, stronger muscles and bones, better weight management, and reduced stress and anxiety. It also highlights the importance of finding an exercise routine that is enjoyable and sustainable.

Translation: ChatGPT can translate text from one language to another. Example:

Translate this sentence from English to Spanish. "The sky is blue."
ChatGPT: El cielo es azul.

Tools&Libraries support LLMs

a. Popular NLP libraries, such as TensorFlow, PyTorch, spaCy, Hugging Face Transformers, AllenNLP,OpenAI GPT-3 API, AllenAI's ELMO, spaCy Transformers etc provide tools for working with large language models. These libraries allow for easy fine-tuning and deployment of models.

b. Some large language models, such as GPT-3, provide APIs for access to their models. This can simplify the process of integrating a large language model into a real-world application.

Challenges and limitations with LLMs

- One of the main challenges with LLMs is the potential for biased or offensive language, as the models learn from the patterns found in the training data.
- Unethical considerations, such as gender and racial biases.
- Amount of computational resources needed to train and run LLMs, which can be expensive and energy-intensive.
- Handling out-of-vocabulary words
- Improving interpretability. While large language models have shown impressive performance on a variety of NLP tasks, they may not perform as well on specific tasks, such as those that require a deeper understanding of the underlying context.

Applications of LLMs

- LLMs are used in a wide range of applications, **including language translation, chatbots, content creation, and text summarization**.

- They can also be used to improve search engines, voice assistants, and virtual assistants.

Future of LLMs

The future of LLMs is promising, with ongoing research focused on improving their accuracy, reducing bias, and making them more accessible and energy-efficient.

As the demand for AI-driven applications continues to grow, LLMs will play an increasingly important role in shaping the future of human-machine interaction.

Ashish Patel • Principal Research Scientist • ashishpatel.ce.2011@gmail.com   **Abonia Sojasingarayar** • Machine Learning Scientist • aboniaa@gmail.com   *Updated:* 2023-02

Large Language Models ：： CHEAT SHEET

Transformer Architecture
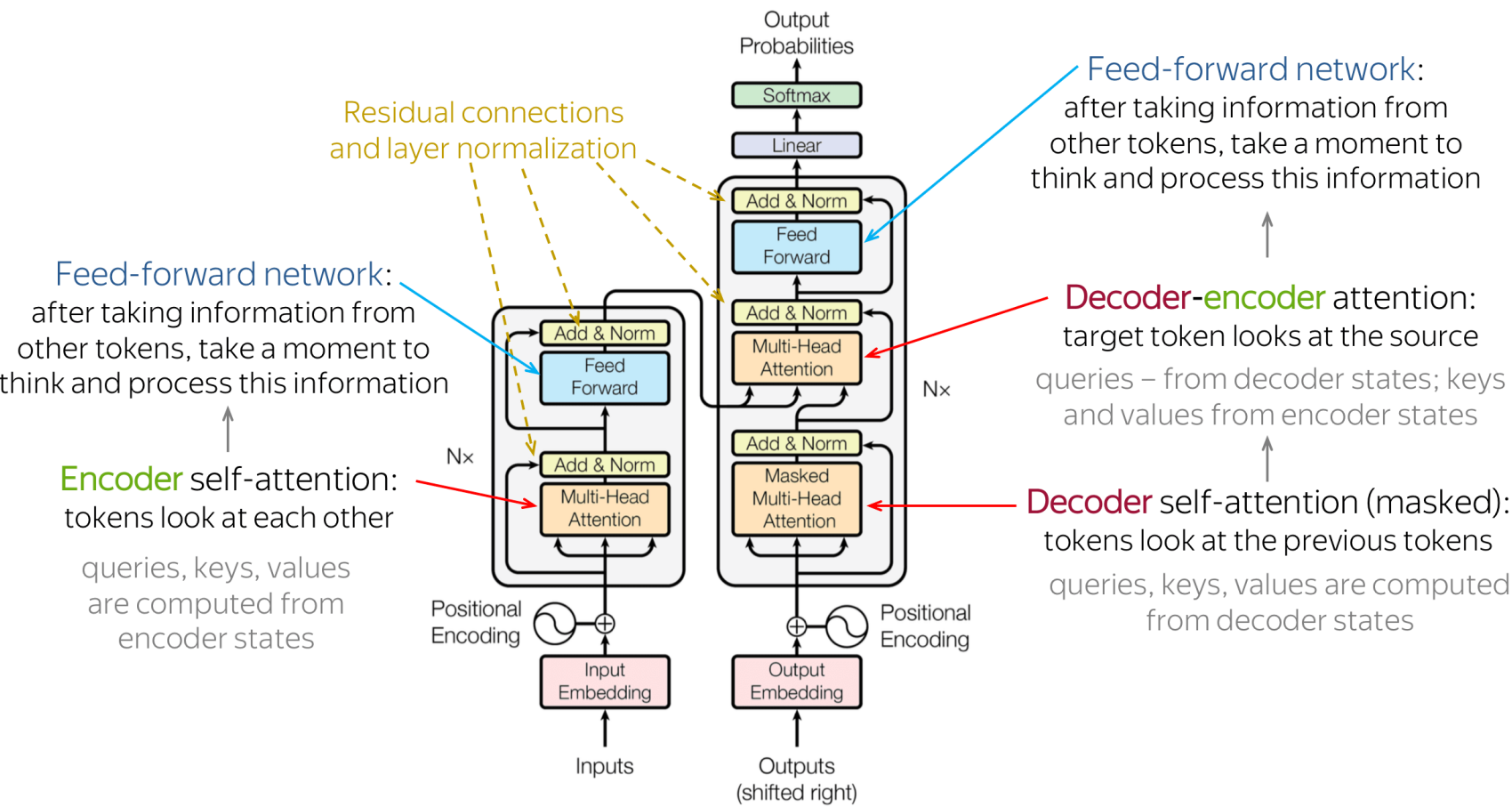
GPT Architecture

Feed-forward network:
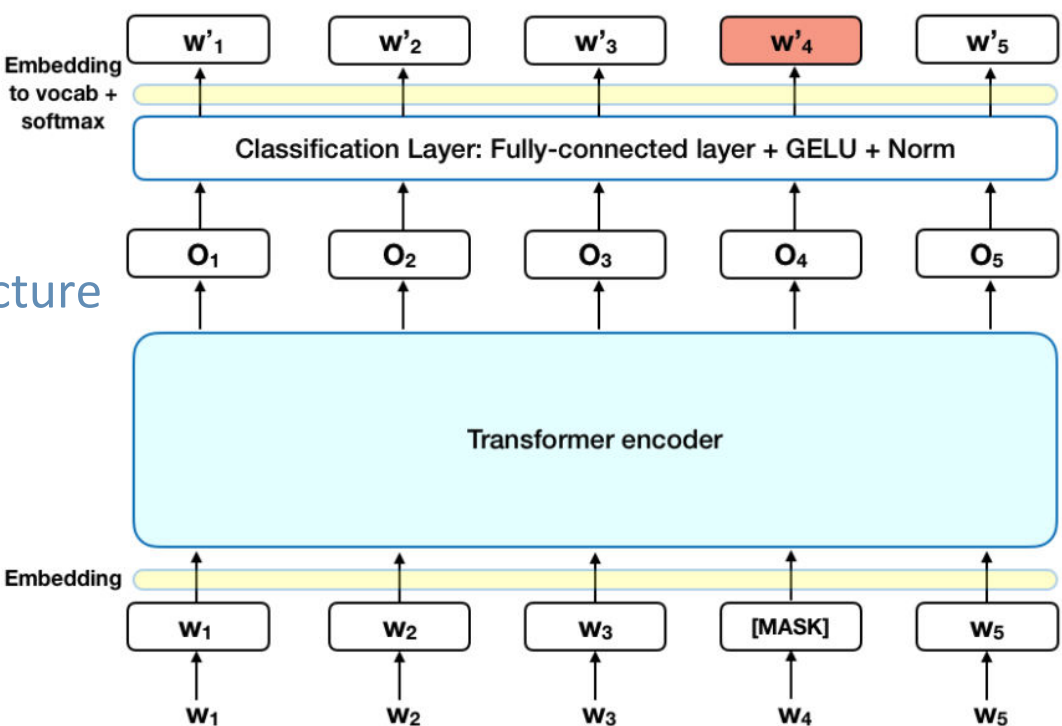after taking information from
other tokens, take a moment to
think and process this information

Residual connections
and layer normalization

Feed-forward network:
after taking information from
other tokens, take a moment to
think and process this information

Decoder-encoder attention:
target token looks at the source
queries – from decoder states; keys
and values from encoder states

Encoder self-attention:
tokens look at each other

queries, keys, values
are computed from
encoder states

Decoder self-attention (masked):
tokens look at the previous tokens

queries, keys, values are computed
from decoder states

BERT Architecture

Ashish Patel • Principal Research Scientist • ashishpatel.ce.2011@gmail.com   Abonia Sojasingarayar • Machine Learning Scientist • aboniaa@gmail.com   Updated: 2023-02