Evaluating Large Language Models (LLMs): Set of Metrics for Accurate Assessment

Objectives

Large Language Models *(LLMs)*are a type of artificial intelligence model that can generate human-like text. They are trained on large amounts of text data and can be used for a variety of natural language processing tasks, such as language translation, question answering, and text generation.

Evaluating LLMs is important to ensure that they are performing well and generating high-quality text. This is especially important for applications where the generated text is used to make decisions or provide information to users.

LLM -Models	License	Reference
Big Code/Starcoder2-7b	bigcode-openrail-m	Hugging Face Website
Google/Gemma-7b	gemma-terms-of-use (other)	Hugging Face Website
Open-Interpreter-Chat	OpenAI	Hugging Face Website
Mistral-7b	Mistral AI	Hugging Face Website
Meta/Codellama-13b	llama2	Hugging Face Website
Yi-34B	Apache-2.0, Model License	Hugging Face Website

Note:We have picked smallest model because of colab. any model below <7b is for mobile/real time applications ,not for accuracy.

Standard Set of Metrics for Evaluating LLMs

There are several standard metrics for evaluating LLMs, including ***perplexity, accuracy, F1-score, ROUGE score, BLEU score, METEOR score, question answering metrics, sentiment analysis metrics, named entity recognition metrics, and contextualized word embeddings.*** These metrics help in assessing LLM performance by measuring various aspects of the generated text, such as fluency, coherence, accuracy, and relevance.

Types of Metrics	Descriptions	Formula/Logic	Sample Results
Perplexity	Perplexity is a measure of how well a language model predicts a sample of text. It is calculated as the inverse probability of the test set normalized by the number of words.	Perplexity can be calculated using the following formula: perplexity = $2^{(-\log P(w_1,w_2,...,w_n)/n)}$, where $P(w_1,w_2,...,w_n)$ is the probability of the test dataset and n is the number of words in the test dataset.	The test set consists of 1000 words, and the language model assigns a probability of 0.001 to each word. The perplexity of the language model on the test set is 2^(-

			log(0.001*1000)/1000) = 31.62.
Accuracy	Accuracy is a measure of how well a language model makes correct predictions. It is calculated as the number of correct predictions divided by the total number of predictions.	Accuracy = (number of correct predictions) / (total number of predictions)	Test the model on a set of 100 images, of which 80 are A and 20 are B. The model correctly classifies 75 A and 15 B. The accuracy of the model is (75+15)/(80+20) = 0.9.
F1-Score	F1-score is a measure of a language model's balance between precision and recall. It is calculated as the harmonic mean of precision and recall.	F1-score = 2 (precision recall) / (precision + recall)	Identify Spam/Ham mails
ROUGE Score	ROUGE score is a measure of how well a language model generates text that is similar to reference texts. It is commonly used for text generation tasks such as summarization and paraphrasing.	ROUGE score can be calculated using various methods, such as ROUGE-N, ROUGE-L, and ROUGE-W. These methods compare the generated text to one or more reference texts and calculate a score based on the overlap between them.	The ROUGE score of the model is calculated based on the overlap between the generated summaries and the actual summaries.
BLEU Score	BLEU score is a measure of how well a language model generates text that is fluent and coherent.	BLEU score can be calculated by comparing the generated text to one or more reference texts and calculating a score based on the n-gram overlap between them.	Test the model on a set of 100 images, and the generated captions are compared to the actual captions of the images
METEOR Score	METEOR score is a measure of how well a language model generates text that is accurate and relevant.	METEOR score can be calculated by comparing the generated text to one or more reference texts and calculating a score based on the harmonic mean of precision and recall.	The METEOR score of the model is calculated based on the harmonic mean of precision and recall.
Q&A Metrics	Question answering metrics are used to evaluate the ability of a language model to provide correct answers to questions.	Question answering metrics can be calculated by comparing the	Test the model on a set of 100 questions, and

	Common metrics include accuracy, F1-score, and Macro F1-score.	generated answers to one or more reference answers and calculating a score based on the overlap between them.	the generated answers are compared to the actual answers. The accuracy, F1-score, and Macro F1-score of the model are calculated based on the overlap between the generated answers and the actual answers.
Sentiment Analysis	Sentiment analysis metrics are used to evaluate the ability of a language model to classify sentiments correctly. Common metrics include accuracy, weighted accuracy, and macro F1-score.	Sentiment analysis metrics can be calculated by comparing the generated sentiment labels to one or more reference labels and calculating a score based on the overlap between them.	Test the model on a set of 100 reviews, and the generated sentiment labels are compared to the actual labels. The accuracy, weighted accuracy, and macro F1-score of the model are calculated based on the overlap between the generated labels and the actual labels.
Named Entity Recognition	Named entity recognition metrics are used to evaluate the ability of a language model to identify entities correctly. Common metrics include accuracy, precision, recall, and F1-score.	metrics can be calculated by comparing the generated entity labels to one or more reference labels and calculating a score based on the overlap between them.	The accuracy, precision, recall, and F1-score of the model are calculated based on the overlap between the generated labels and the actual labels.
Contextualization Word Embeddings	Contextualized word embeddings are used to evaluate the ability of a language model to capture context and meaning in word representations. They are generated by training the language model to predict the next word in a sentence given the previous words.	Calculating a score based on the similarity between generated embeddings to one or more reference embeddings	The evaluation can be done using various methods, such as cosine similarity and Euclidean distance

Evaluation Results

1.Mistral-7B-Chat Model

Here is the Mistral-7B model which has chat template(If the model does not have chat template ,we need to google)

The below screenshot reads, fine-tuned prompt for various tasks based on the LLM benchmark

This example has questions and multiple choices.

category	question	A	B	C	D	right answer
anatomy	Which of the following is the body cavity that contains the pituitary gland?	Abdominal	Cranial	Pleural	Spinal	B
computer-security	The ___________ is anything which your search engine cannot search.	Haunted web	World Wide Web	Surface web	Deep Web	D
machine-learning	What is the dimensionality of the null space of the following matrix? A = [[1, 1, 1], [1, 1, 1], [1, 1, 1]]	0	1	2	3	C

Here the example has questions and multiple answers

```
[17] prompt = "what is the value of the equation: \n\n 2+2-3. \n\n Choices a) 7 b) 1 c) 2. \n\n The answer is"

[ ]

messages = [
    {"role": "system", "content": "You are a helpful assistant."},
    {"role": "user", "content": prompt}
]
formatted_messages = tokenizer.apply_chat_template(messages, tokenize=False, add_generation_prompt=True)

print(formatted_messages)

<|im_start|>system
You are a helpful assistant.<|im_end|>
<|im_start|>user
what is the value of the equation:

 2+2-3.

 Choices a) 7 b) 1 c) 2.

 The answer is<|im_end|>
<|im_start|>assistant
```

I have created something similar math question and expecting right answer from multiple choice. The LLM picks the correct answer

```
[14] prompt = "what is the value of the equation: \n\n 2+2-3. \n\n Choices a) 7 b) 1 c) 2. \n\n The answer is"
     #prompt = "what are the symptoms of diabetes?"
     #prompt = "{Name: XYZ, healthstats: {weight: 80kg, height:175cms, blood pressure: 120/90, fasting glucose: 300, hemoglobin count:4000 } } \n\nAm I prediabetic ?"

[ ] Start coding or generate with AI.

messages = [
    {"role": "system", "content": "You are a helpful healthcare assistant."},
    {"role": "user", "content": prompt}
]
formatted_messages = tokenizer.apply_chat_template(messages, tokenize=False, add_generation_prompt=True)

print(formatted_messages)

<|im_start|>system
You are a helpful healthcare assistant.<|im_end|>
<|im_start|>user
what is the value of the equation:

 2+2-3.

 Choices a) 7 b) 1 c) 2.

 The answer is<|im_end|>
<|im_start|>assistant
```

```
print(response)

The value of the equation 2 + 2 - 3 is 1. Therefore, the correct choice is b) 1.
```

Double-click (or enter) to edit

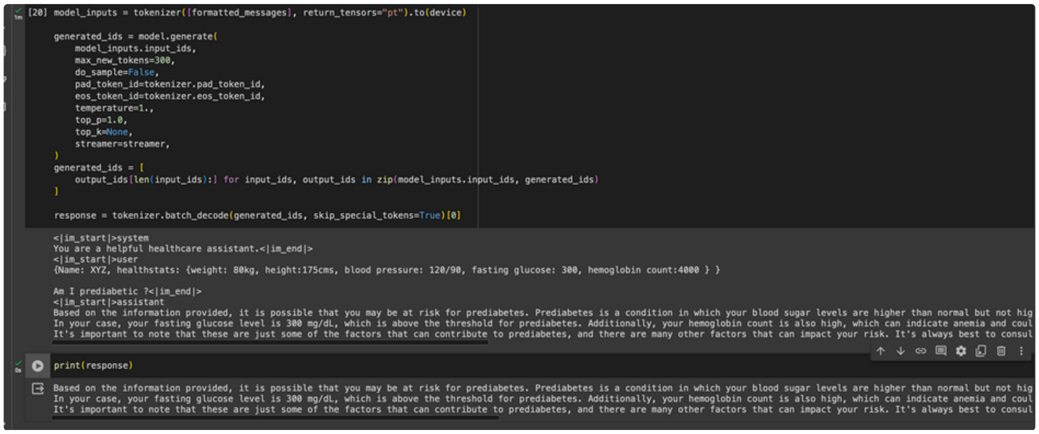
The Correct Answer is - b) 1

LLM Accuracy Score - 10/10

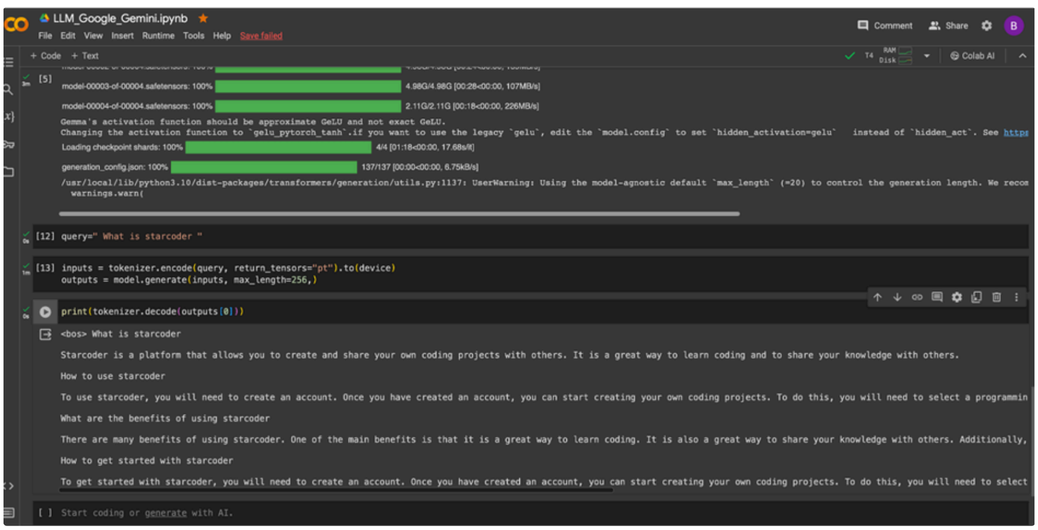
Prompt # 2 - What are the symptoms of diabetes?

```
#prompt = "what is the value of the equation: \n\n 2+2-3. \n\n Choices a) 7 b) 1 c) 2. \n\n The answer is"
prompt = "what are the symptoms of diabetes?"
#prompt = "{Name: XYZ, healthstats: {weight: 80kg, height:175cms, blood pressure: 120/90, fasting glucose: 300, hemoglobin count:4000 } } \n\nAm I prediabetic ?"

[ ] Start coding or generate with AI.

[11] messages = [
    {"role": "system", "content": "You are a helpful healthcare assistant."},
    {"role": "user", "content": prompt}
]
formatted_messages = tokenizer.apply_chat_template(messages, tokenize=False, add_generation_prompt=True)

print(formatted_messages)

<|im_start|>system
You are a helpful healthcare assistant.<|im_end|>
<|im_start|>user
what are the symptoms of diabetes?<|im_end|>
<|im_start|>assistant
```

Prompt#3 - Am I Pre-diabetic? (verified by given test data)



2. Google/Gemma-7b

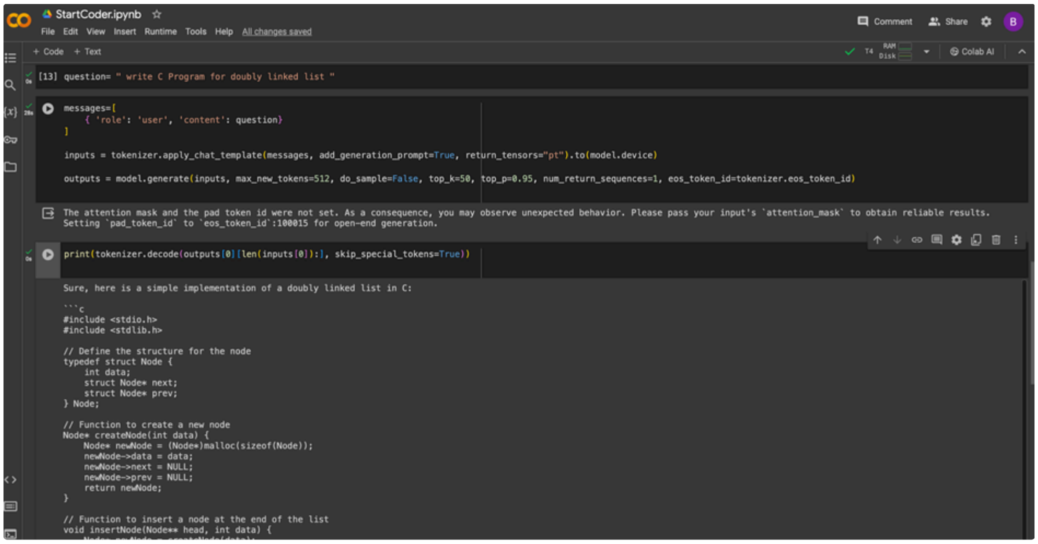


Accuracy Score - 100%

3. Open- Interpreter

TBD

4.StartCoder/DeepSeekAI



Code Generator AI LLM - StarCoder & DeepseekAI

5.Yi

TBD

Conclusion

It is important to choose the appropriate metrics for specific tasks to ensure that the LLM is evaluated accurately and comprehensively.