

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/Starcode2-7b	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: $\text{perplexity} = 2^{(-\log P(w_1, w_2, \dots, w_n) / n)}$ where $P(w_1, w_2, \dots, w_n)$ 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 $2^{(-\log 0.001)}$.

			$\log(0.001 \times 1000) / 1000 = 31.62$.
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 = $2 \times (\text{precision} \times \text{recall}) / (\text{precision} + \text{recall})$	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? $A = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	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
```

```
[13] print(response)

The symptoms of diabetes can vary depending on the type of diabetes, but some common signs and symptoms include:
• Increased thirst and frequent urination
• Fatigue or weakness
• Blurred vision
• Hunger or unexplained weight loss
• Nausea or vomiting
• Slow healing wounds
• Frequent infections
• Tingling or numbness in the hands or feet
```

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

```
[20] model_inputs = tokenizer(formatted_messages, return_tensors="pt").to(device)

generated_ids = model.generate(
    model_inputs.input_ids,
    max_new_tokens=300,
    do_sample=False,
    pad_token_id=tokenizer.pad_token_id,
    eos_token_id=tokenizer.eos_token_id,
    temperature=1.,
    top_p=1.0,
    top_k=None,
    streamer=streamer,
)

generated_ids = [
    output_ids[len(input_ids):] for input_ids, output_ids in zip(model_inputs.input_ids, generated_ids)
]

response = tokenizer.batch_decode(generated_ids, skip_special_tokens=True)[0]

<[im_start]>system
You are a helpful healthcare assistant.<[im_end]>
<[im_start]>user
(Name: XYZ, healthstats: (weight: 80kg, height:175cms, blood pressure: 120/90, fasting glucose: 300, hemoglobin count:4000 ) )

Am I pre-diabetic ?<[im_end]>
<[im_start]>assistant
Based on the information provided, it is possible that you may be at risk for prediabetes. Prediabetes is a condition in which your blood sugar levels are higher than normal but not high enough to be diagnosed as diabetes. In your case, your fasting glucose level is 300 mg/dL, which is above the threshold for prediabetes. Additionally, your hemoglobin count is also high, which can indicate anemia and could be a sign of a more serious condition. It's important to note that these are just some of the factors that can contribute to prediabetes, and there are many other factors that can impact your risk. It's always best to consult with a healthcare professional for a thorough evaluation and personalized advice.

print(response)

Based on the information provided, it is possible that you may be at risk for prediabetes. Prediabetes is a condition in which your blood sugar levels are higher than normal but not high enough to be diagnosed as diabetes. In your case, your fasting glucose level is 300 mg/dL, which is above the threshold for prediabetes. Additionally, your hemoglobin count is also high, which can indicate anemia and could be a sign of a more serious condition. It's important to note that these are just some of the factors that can contribute to prediabetes, and there are many other factors that can impact your risk. It's always best to consult with a healthcare professional for a thorough evaluation and personalized advice.
```

2. Google/Gemma-7b

```
LLM_Google_Gemini.ipynb
File Edit View Insert Runtime Tools Help Save failed

[15] model-0003-of-0004.safetensors: 100% 4.98GiB [00:28<00:00, 107MiB/s]
model-0004-of-0004.safetensors: 100% 2.11GiB [00:18<00:00, 226MiB/s]
Gemma's activation function should be approximate GeLU and not exact GeLU.
Changing the activation function to "gelu_pytorch_tanh". If you want to use the legacy "gelu", edit the "model.config" to set "hidden_activation=gelu" instead of "hidden_act". See https://huggingface.co/google/gemma-7b
Loading checkpoint shards: 100% 4/4 [01:18<00:00, 17.6MiB/s]
generation.config.json: 100% 137/137 [00:00<00:00, 6.79MiB/s]
/usr/local/lib/python3.10/dist-packages/transformers/generation/utils.py:1137: UserWarning: Using the model-agnostic default 'max_length' (=>20) to control the generation length. We recommend setting 'max_new_tokens' to achieve the desired behavior.
warnings.warn(

[12] query=" What is starcoder "

[13] inputs = tokenizer.encode(query, return_tensors="pt").to(device)
outputs = model.generate(inputs, max_length=256,)

print(tokenizer.decode(outputs[0]))

<bos> What is starcoder

Starcoder is a platform that allows you to create and share your own coding projects with others. It is a great way to learn coding and to share your knowledge with others.

How to use starcoder

To use starcoder, you will need to create an account. Once you have created an account, you can start creating your own coding projects. To do this, you will need to select a programming language and a template.

What are the benefits of using starcoder

There are many benefits of using starcoder. One of the main benefits is that it is a great way to learn coding. It is also a great way to share your knowledge with others. Additionally, it allows you to collaborate with other developers and get feedback on your code.

How to get started with starcoder

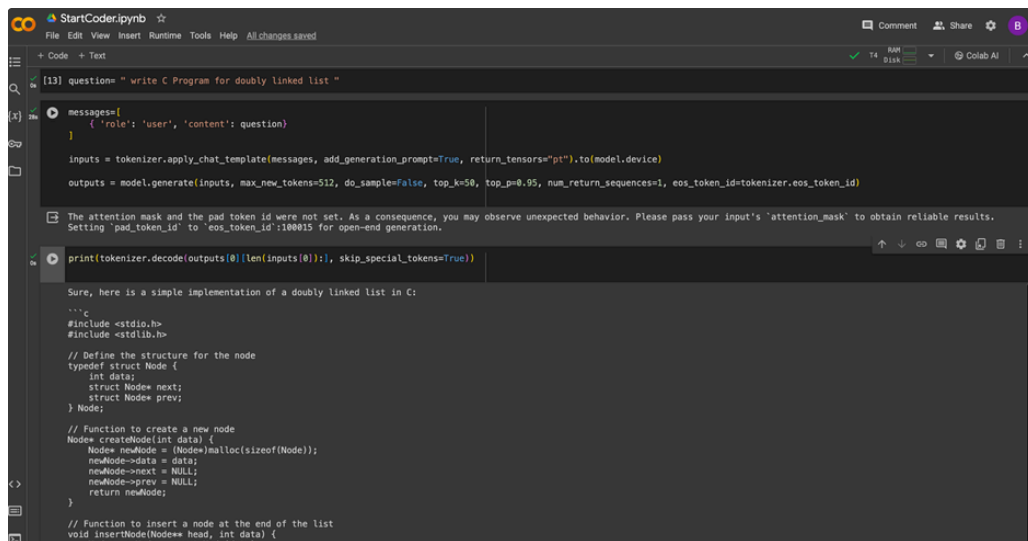
To get started with starcoder, you will need to create an account. Once you have created an account, you can start creating your own coding projects. To do this, you will need to select a programming language and a template.
```

Accuracy Score - 100%

3. Open- Interpreter

TBD

4.StartCoder/DeepSeekAI



The screenshot shows a web-based code editor titled "StarCoder.ipynb". The interface includes a menu bar (File, Edit, View, Insert, Runtime, Tools, Help) and a toolbar with icons for commenting, sharing, and settings. The main area displays a Jupyter Notebook cell with the following content:

```
[13] question= " write C Program for doubly linked list "
```

```
Messages=[
  { 'role': 'user', 'content': question }
]

inputs = tokenizer.apply_chat_template(messages, add_generation_prompt=True, return_tensors="pt").to(model.device)
outputs = model.generate(inputs, max_new_tokens=512, do_sample=False, top_k=50, top_p=0.95, num_return_sequences=1, eos_token_id=tokenizer.eos_token_id)

# The attention mask and the pad token id were not set. As a consequence, you may observe unexpected behavior. Please pass your input's 'attention_mask' to obtain reliable results.
# Setting 'pad_token_id' to 'eos_token_id':100015 for open-end generation.

print(tokenizer.decode(outputs[0][len(inputs[0]):], skip_special_tokens=True))
```

The output of the cell is a C program for a doubly linked list:

```
Sure, here is a simple implementation of a doubly linked list in C:

'''C
#include <stdio.h>
#include <stdlib.h>

// Define the structure for the node
typedef struct Node {
    int data;
    struct Node* next;
    struct Node* prev;
} Node;

// Function to create a new node
Node* createNode(int data) {
    Node* newNode = (Node*)malloc(sizeof(Node));
    newNode->data = data;
    newNode->next = NULL;
    newNode->prev = NULL;
    return newNode;
}

// Function to insert a node at the end of the list
void insertNode(Node** head, int data) {
```

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.