Introduction to Evaluating LLMs and Their Applications

What is Evaluation?

- Systematic assessment and measurement of the performance of LLMs and their applications.

Traditional Evaluation Metrics for NLP Tasks

Since LLMs are solutions to NLP problems, let’s start with analyzing traditional evaluation metrics.

- BLEU
    - Assesses the quality of text by comparing it to reference texts.
    - Relies on n-grams.
- ROUGE
    - Unlike BLEU score (often used in machine translation), ROUGE leans more towards recall.

Limitations of Traditional Evaluation Metrics for NLP Tasks

- Focus on Surface-Level Similarity: Metrics like BLEU for machine translation heavily emphasize matching n-grams (word sequences) with human-created references. They don't fully capture the nuances of meaning, fluency, or whether a text actually fulfills the intended purpose.

- Limited Contextual Understanding: These metrics sometimes struggle to evaluate text in situations where context matters heavily, like in dialogue systems or creative text generation.

- Reliance on Reference Data: Traditional metrics often rely heavily on having high-quality reference text like human translations or summaries. Creating these references is expensive and time-consuming.

Non-Traditional Evaluation Metrics for NLP

These metrics rely on language models or embeddings to analyze the generated text.

- BERTScore
- BLEURT

BERTScore

BERTScore uses BERT to encode the reference text and the generated text. Then, embeddings are compared using cos similarity. Finally, a score is generated based on the similarity.

BLEURT

A novel metric for evaluating the quality of text generated by AI models, primarily in Natural Language Generation tasks.

- Relies on a language model to score the generated text
- It does not need reference data to score the generated text since it is trained on a dataset of text pairs with human ratings

Evaluating Large Language Models

LLMs are primarily evaluated on open task-specific datasets to analyze their capabilities in doing a variety of tasks like summarisation, open book question answering, etc.

Some benchmarks are:

- MMLU
- ARC
- HellaSwag

Evaluating Language Models Using Benchmarks

The benchmarks are simply a dataset of questions. Questions in the data set are asked to the LLM in test. Then, the answers from the LLM are graded. Traditional or non-traditional evaluation methods to check the correctness of answers.

LLM Assisted Evaluation Metrics

Some evaluation metrics make use of LLMs to evaluate LLMs.

- GPTScore
- LLM-as-a-Judge

GPTScore

GPTScore is a unsupervised evaluation method in the sense that it does not require reference text to score the generated text.

This method requires an evaluation protocol which consists of a task specification, aspect specification, exemplars, and the question.

The answers generated from the LLM are fed into another LLM to score the answers.

LLM-as-a-Judge

LLM-as-a-Judge is reminiscent of the GPTScore in the sense that it also makes use of an LLM to grade the answers.

It has several different evaluation methods.

1. Single Answer Grading
2. Reference-Guided Grading
3. Pairwise Comparison

LLM Evaluations in Practice

Let’s learn about how these methods are applied in practice and which frameworks are available for evaluation.

Evaluating Generated Texts

Langchain framework provides a evaluation package. This package contains a “String Evaluator” subpackage. This subpackage can:

- Compare answers generated by LLMs with a reference answer
- Score a text generated by an LLM

Evaluating RAG Applications

There is a framework called auto-evaluator that is publicly available on Github. Auto-evaluator framework is created to evaluate the performance of RAG applications.

It makes use of a dataset to determine how relevant was the retrieval and how relevant was the answer.

RAGAS is another popular framework for evaluating RAG applications.

Evaluating Agents

Langchain has a trajectory evaluation subpackage that analyzes the trajectory of the actions taken by an Agent.