Evaluation Metrics for NLP Models

Learning Objective:

- Introduce essential metrics for evaluating generated text by generative NLP models.

Estimated reading time: 30 minutes

In this reading, you will learn about key metrics that can be used to evaluate the quality of text generated by generative AI models.

Introduction

Natural language processing (NLP) models have witnessed remarkable progress, particularly with the adoption of deep learning techniques. These models have advanced significantly, ranging from chatbots that engage in human-like conversations to language translation systems. However, assessing the quality and performance of these models in generating text presents a unique challenge. Fortunately, several evaluation metrics have been developed to address this challenge. In this article, you will delve into four essential evaluation metrics used to assess the performance of NLP models in generating text:

■ Perplexity

Perplexity is a commonly used metric for evaluating language models. It measures how well a model predicts a sequence of words. The lower the perplexity, the better the model's ability to predict the next word accurately in a given context. Perplexity is derived from the concept of entropy, which quantifies the uncertainty in predicting the next word. A lower perplexity indicates that the model better understands the underlying language patterns.

■ ROUGE (Recall-Oriented Understudy for Gisting Evaluation)

ROUGE is a set of metrics commonly used for evaluating text summarization systems. It measures the overlap between the generated summary and one or more reference summaries. ROUGE calculates various metrics, such as ROUGE-N, which measures the n-gram overlap, and ROUGE-L, which measures the longest common subsequence. Higher ROUGE scores indicate better summarization quality.

■ BLEU (Bilingual Evaluation Understudy)

BLEU is a metric primarily used for evaluating machine translation systems. It compares the generated translation with one or more reference translations and assigns a score based on the degree of overlap. BLEU measures the precision of the generated translation by counting the number of n-grams (contiguous sequences of n words) that appear in both the generated and reference translations. A higher BLEU score indicates a better translation quality.

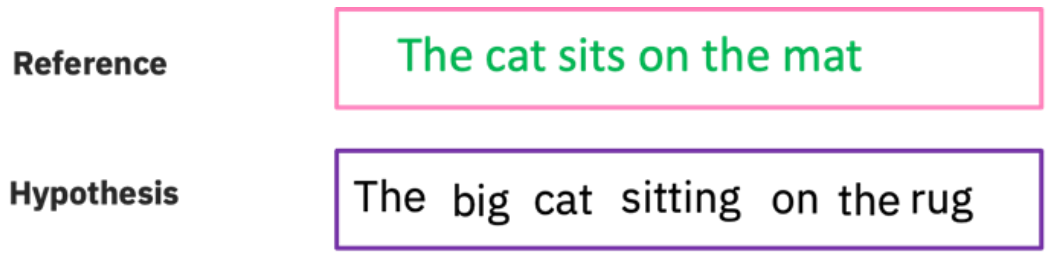
■ METEOR (Metric for Evaluation of Translation with Explicit ORdering)

METEOR is another metric used for evaluating machine translation systems. It considers multiple aspects of translation quality, including precision, recall, and alignment. METEOR incorporates various matching criteria, such as exact word matches, synonymy, stemming, and reordering of words. It also considers the use of paraphrases and recognizes partial matches. A higher METEOR score indicates better translation quality.
These four evaluation metrics provide valuable insights into different aspects of NLP model performance. By leveraging these metrics, researchers and practitioners can better understand the strengths and weaknesses of NLP models, enabling them to make informed decisions when developing and improving these systems.

ROUGE-N

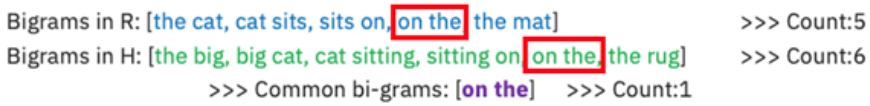
ROUGE-N is a metric used to evaluate the quality of summaries or generated text by comparing them to human-produced reference text. It measures the number of matching n-grams between the model-generated text (referred to as "Hypothesis" or "H") and the human-produced reference text (referred to as "Reference" or "R").

Let's do an example:



Identifying matching n-grams

To compute ROUGE-N, the first step is to identify the matching n-grams between the Reference and Hypothesis sentences. These are the bigrams that appear in both the Hypothesis and Reference.



ROUGE-2 precision

ROUGE-2 precision is computed by taking the ratio of the number of common bigrams in the hypothesis (that is, bigrams appearing in both hypothesis and reference, such as "on the") to the total number of bigrams in the hypothesis. Precision measures the accuracy of the generated text in terms of capturing the correct bigrams.

$$\frac{Number\ of\ matching\ bigrams}{Number\ of\ bigrams\ in\ H} = \frac{1}{6}$$

ROUGE-2 recall

ROUGE-2 recall is computed by taking the ratio of the number of common bigrams to the total number of bigrams in the reference text. Recall measures the completeness of the generated text by considering how many of the important bigrams from the reference text are included.

$$\frac{Number\ of\ matching\ bigrams}{Number\ of\ bigrams\ in\ R} = \frac{1}{5}$$

ROUGE-2 F1-score

The ROUGE-2 F1-score is obtained using the standard F1-score formula, which combines precision and recall. The F1-score is a harmonic mean of precision and recall. It provides a single value that represents the overall performance of the generated text in terms of capturing the correct and important bigrams from the reference text.

$$\boldsymbol{F1 - Score = \frac{2 \times (Precision \times Recall)}{Precision + Recall} = 0.18}$$

ROUGE-L and Longest Common Subsequence (LCS)

ROUGE-L measures the similarity between the model output and the reference text by considering the longest subsequence of words (not necessarily consecutive) that are shared between both. The LCS represents the longest sequence of words appearing in the same order in the model output and the reference text.

Identifying the LCS

To compute ROUGE-L, the first step is to identify the LCS between the model output (H) and the reference text (R). In the given example, the LCS is the 4-gram "the cat on the." It is important to note that the words in the LCS do not have to be consecutive.

ROUGE-L precision

ROUGE-L precision is computed by taking the ratio of the length of the LCS to the number of unigrams (individual words) in the model output (H). Precision measures the accuracy of the model output in terms of capturing the shared words between the model output and the reference text.

$$\frac{Length\ of\ LCS}{Number\ of\ unigrams\ in\ H} = \frac{4}{7}$$

ROUGE-L recall:

ROUGE-L recall is computed by taking the ratio of the length of the LCS to the number of unigrams in the reference text (R). Recall measures the completeness of the model output by considering how many of the shared words from the reference text are included.

Here are the calculations for the previous example:

$$\frac{Length\ of\ LCS}{Number\ of\ unigrams\ in\ R} = \frac{4}{6}$$

ROUGE-S and skip-gram matching

ROUGE-S extends the matching capabilities of ROUGE-N and ROUGE-L by considering skip-grams, which involve finding consecutive words from the reference text that appear in the model output, even if other words separate them.

Example of matching with ROUGE-2 and ROUGE-S

To illustrate the difference, consider the 2-gram "the cat." With ROUGE-2, this 2-gram would only match if it appears exactly as "the cat" in the model output (referred to as "H"). However, if the model output contains "the big cat" instead, it would not be considered a match. Conversely, ROUGE-S allows for unigram skipping, meaning that "the cat" would match "the big cat" as well.

Finding common Bi-grams

To compute ROUGE-S, the first step is to find the bi-grams (2-grams) in both the reference text (R) and the model output (H). Considering the skip-gram matching, these bigrams appear in both R and H.

R: [the cat, the sits, cat sits, cat on, sits on, sits the, on the, on mat, the mat]: 9

H: [the big, the cat, big cat, big sitting, cat sitting, cat on, sitting on, sitting the, on the, on rug, the rug]: 11

>>> Common bi-grams: [the cat, cat on, on the]: 3

Computing ROUGE-S precision and recall

ROUGE-S precision is computed by taking the ratio of the number of common bi-grams to the total number of bi-grams in H. ROUGE-S recall is computed by taking the ratio of the number of common bi-grams to the total number of bi-grams in R.

$$\boldsymbol{Precision} = \frac{\mathbf{3}}{\mathbf{11}}, \qquad \boldsymbol{Recall} = \frac{\mathbf{3}}{\mathbf{9}}$$

⇒ **Note:** ROUGE-N, ROUGE-L, and ROUGE-S can be used with multiple references, allowing for a more comprehensive evaluation of the generated text.

BLEU (BiLingual Evaluation Understudy)

Validating the results using the BLEU score is helpful when there is more than one valid translation for a sentence, as you can include many translation versions in the reference list and compare the generated translation with its different versions.

The BLEU (Bilingual Evaluation Understudy) score is a metric commonly used to evaluate the quality of machine-generated translations by comparing them to one or more reference translations. It measures the similarity between the generated and reference translations based on n-gram matching.

Clipped precision is a metric used to determine the proportion of n-grams in a generated translation present in the reference translations. It differs from traditional precision in two significant ways:

- Clipped precision sets an upper limit on matching n-gram counts to prevent inflated scores for translations that repeat words found in the reference. For example, in the case of the generated translation "cat cat cat sits sits sits" and the reference translation "the cat sits on the mat," while traditional precision would yield a score of 5/5, clipped precision would yield a score of 2/5.
- Clipped precision compares each n-gram with the n-grams in all reference translations, counting if it occurs in any references.

Clipped precision is calculated for each n-gram order (1 to N) and then combined using a geometric mean. The clipped precision for a particular n-gram order is calculated as follows:

$$ClippedPrecision_n = \frac{CountClip_n}{Count_n}$$

$CountClip_n$ is the count of n-grams in the generated translation that appears in any reference translation, clipped by the maximum count of that n-gram in any single reference translation.

$Count_n$ is the count of n-grams in the generated translation.

After computing precision for n-grams, a brevity penalty is applied to account for translation length. Finally, the BLEU score is calculated as the weighted geometric mean of the precisions multiplied by the penalty.

$$BP = \begin{cases} 1 & if\ h > r \\ e^{(1-\frac{r}{h})} & if\ h \leq r \end{cases}$$

$$BLEU = BP \times \exp\left(\sum_{n=1}^{N} W_n \log(ClippedPrecision_n)\right)$$

Let's do an example calculation step by step.

Assuming you are only interested in calculating BLEU for unigram, bigram, and trigram, you can count the matching unigrams, bigrams, and trigrams and the number of Hypothesis N-grams.

Reference	Hypothesis
The cat sits on the mat	The big cat sitting on the mat
Unigrams: the, cat, sits, on, the, mat Bigrams: the cat, cat sits, sits on, on the, the mat Trigrams: the cat sits, cat sits on, sits on the, on the mat	Unigrams: the, big, cat, sitting, on, the, mat Bigrams: the big, big cat, cat sitting, sitting on, on the, the mat Trigrams: the big cat, big cat sitting, cat sitting on, sitting on the, on the mat
Common Unigram counts: 5 Common Bigram counts:2 Common Trigram counts:1	Unigram counts: 7 Bigram counts:6 Trigram counts:5

Next, calculate the precisions for each N-gram order:

Precision$_1$: 5/7
Precision$_2$: 2/6
Precision$_3$: 1/5

Then, calculate brevity:

Brevity: min(1, exp(1 - (6/7))) = 1

Finally, calculate the BLEU score with arbitrary weights for n-gram precisions.

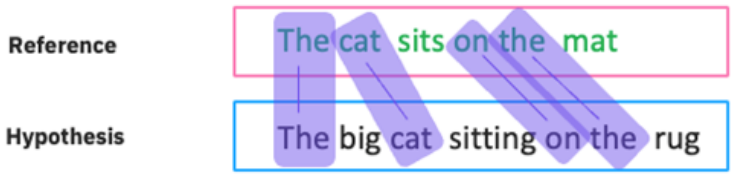
W = {1/2,1/4,1/4}
BLEU: Brevity * exp((1/2) * log(5/7) +(1/4) * log(2/6) +(1/4) * log(1/5))≈ 0.55

METEOR (Metric for Evaluation of Translation with Explicit ORdering)

METEOR compares the words in the system translation to a reference translation. If there are multiple reference translations, the system translation is compared separately to each translation, and the best match is chosen. The first step is creating a word alignment between the system and reference translations.

Counting matching unigrams

Once the word alignment is established, the number of matching individual words (unigrams) between the system translation (H) and the reference translation (R) is counted. This count is denoted as "m."



Precision and recall

Precision and recall are computed to calculate the METEOR score. Precision is calculated by dividing "m" by the total number of unigrams in the system translation (H). In contrast, recall is calculated by dividing "m" by the total number of unigrams in the reference translation (R).

$$Precision = \frac{4}{7}, \qquad Recall = \frac{4}{6}$$

Harmonic mean

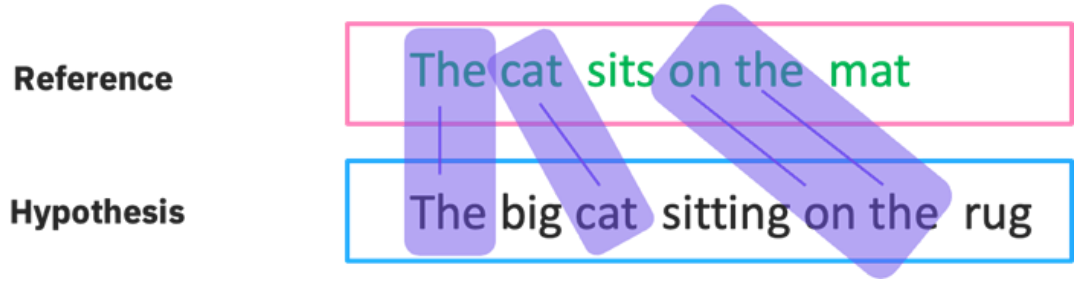
A parameterized average of precision and recall, called the harmonic mean, is computed. The harmonic mean provides a balance between precision and recall and is calculated by taking the reciprocal of the arithmetic mean of the reciprocals of precision and recall.

$$Harmonic\ Mean = \frac{Precision \times Recall}{\alpha \times Precision + (1-\alpha) \times Recall}$$

Assuming $\alpha = 0.5$, therefore, Harmonic Mean = 0.61.

Penalty for word order

To consider the extent to which the matched unigrams in the system and reference translations are in the same word order, METEOR computes a penalty for a given alignment. The sequence of matched unigrams is divided into the fewest possible number of "chunks," where the matched unigrams in each chunk are adjacent and in identical word order in both strings.



Penalty calculation

The number of chunks (denoted as "ch") and the number of matches (denoted as "m") are used to calculate the penalty.

The penalty calculation takes into account the parameter γ, which determines the maximum penalty and is a value between 0 and 1.

Assuming $\gamma = 0.8$:

$$Penalty = \gamma \times \left(\frac{ch}{m}\right)^3 = 0.8 \times \left(\frac{3}{4}\right)^3 = 0.34$$

METEOR score calculation

The METEOR score for the alignment between the system translation and the reference translation is calculated as follows:

$$METEOR = (1 - Penalty) \times Harmonic\ mean = (1 - 0.34) \times 0.61 = 0.41$$

This score represents the similarity between the system and reference translations, considering word matching and word order.

Summary

In this reading, you learned about:

- The four types of metrics that are used for evaluating the quality of text generated by generative models
- The differences between different metrics and their uses in NLP tasks

Skills Network IBM