CS 335: Introduction to Large Language Models

Evaluating Language Models – ROUGE Score

Week 4

Dr Abdul Samad

Dr Faisal Alvi

Lecture Outline

- What is ROGUE ?
- ROGUE Variants
- ROUGE-N
- ROUGE-L
- ROUGE-S

Recall-Oriented Understudy for Gisting Evaluation (ROUGE)

ROUGE score stands for Recall-Oriented Understudy for Gisting Evaluation. ROUGE is an evaluation metric used to assess the quality of NLP tasks such as text summarization and machine translation. Unlike BLUE, the ROUGE uses both recall and precision to compare the model generated summaries (candidate) against human generated summaries (reference).

ROGUE Variants

ROUGE is a set of metrics, rather than just one. The main ones that will be discussed are:

- ROUGE-N
- ROUGE-L
- ROUGE-S

ROUGE-N

ROUGE-N measures the number of matching n-gram between the model generated text and a reference. With ROUGE-N, the N represents the n-gram that we are using. For example, ROUGE-1 would be measuring the matching-rate of unigrams between our model output and reference. ROUGE-2 and ROUGE-3 would use bigrams and trigrams respectively.

Once we have decided which N to use, we can calculate the ROUGE recall, precision or F1 score.

ROUGE-N Recall, Precision, F1-Score

Recall

$$\frac{\text{number of correct predicted words}}{\text{number of total actual words}}$$

Precision

$$\frac{\text{number of correct predicted words}}{\text{number of total predicted words}}$$

F1-Score

$$2 * \frac{precision * recall}{precision + recall}$$

ROUGE-1 Example

Reference Text: the fox jumps

Candidate Text: the hello a cat dog fox jumps

Recall

$$\frac{3}{3} = 100\%\ recall$$

Precision

$$\frac{3}{7} = 43\%\ precision$$

F1-Score

$$2\ *\ \frac{0.43\ *1.0}{0.43 + 1.0} = 60\%\ f1\ score$$

ROUGE-L

ROUGE-L measures the longest common subsequence (LCS) between the candidate text and reference. All this means is that we count the longest sequence of tokens that is shared between both.

One advantage of using LCS is that it does not require consecutive matches but in-sequence matches that reflect sentence level word order as n-grams.

However, LCS suffers one disadvantage that it only counts the main in-sequence words

ROUGE-L Recall, Precision, F1-Score

Recall

$$\frac{LCS}{Number\ of\ total\ actual\ words}$$

Precision

$$\frac{LCS}{Number\ of\ total\ predicted\ words}$$

F1-Score

$$2 * \frac{precision * recall}{precision + recall}$$

ROUGE-L Example

Reference Text: the fox jumps

Candidate Text: the hello a cat dog fox jumps

Recall

$$\frac{2}{3} = 66\%\ recall$$

Precision

$$\frac{2}{7} = 29\%\ precision$$

F1-Score

$$2\ *\ \frac{0.29\ *0.66}{0.29 + 0.66} = 40\%\ f1\ score$$

ROUGE-L Example

Reference Text: police killed the gunman

Candidate Text 1: police kill the gunman

Candidate Text 2: the gunman kill police

The ROGUE-2 score for both these candidates is 50%.

Recall

Candidate Text 1 : $\frac{3}{4} = 75\%\ recall$ Candidate Text 2 : $\frac{2}{4} = 50\%\ recall$

Precision

Candidate Text 1 : $\frac{3}{4} = 75\%\ precision$ Candidate Text 2 : $\frac{2}{4} = 50\%\ precison$

F1-Score

Candidate Text 1 : $\frac{3}{4} = 75\%\ f1 - score$ Candidate Text 2 : $\frac{2}{4} = 50\%\ f1 - score$

ROUGE-S

ROUGE-S refers to the skip-bigram concurrence metric. Using the skip-bigram metric allows us to search for consecutive words from the reference text, that appear in the candidate text but are separated by one-or-more words.

One advantage of skip-bigram vs. LCS is that it does not require consecutive matches but is still sensitive to word order.

Comparing skip-bigram with LCS, skip-bigram counts all in-order matching word pairs while LCS only counts one longest common subsequence.

For example, the sentence “police killed the gunman” has the following 6 skip-bigrams: (“police killed”, “police the”, “police gunman”, “killed the”, “killed gunman”, “the gunman”)

ROUGE-S Recall, Precision, F1-Score

Recall

$$\frac{\text{number of correct predicted } skip-bigrams}{\text{number of total } actual\ skip-bigrams}$$

Precision

$$\frac{\text{number of correct predicted } skip-bigrams}{\text{number of total predicted } skip-bigrams}$$

F1-Score

$$2 * \frac{precision * recall}{precision + recall}$$

ROUGE-S Example

Reference Text: the fox

Candidate Text: the brown fox jumps

Recall

$$\frac{1}{1} = 100\%\ recall$$

Precision

$$\frac{1}{3} = 33\%\ precision$$

F1-Score

$$2\ *\ \frac{0.33 * 1.0}{0.33 + 1} = 50\%\ f1\ score$$

ROUGE-S Example

Reference Text: police killed the gunman

(“police killed”, “police the”, “police gunman”, “killed the”, “killed gunman”, “the gunman”)

Candidate Text 1: police kill the gunman

(“police kill”, “police the”, “police gunman”, “kill the”, “kill gunman”, “the gunman”)

Candidate Text 2: the gunman kill police

(“the gunman”, “the kill”, “the police”, “gunman kill”, “gunman police”, “kill police”)

Recall

Candidate Text 1 : $\frac{3}{6} = 50\%\ recall$ Candidate Text 2 : $\frac{1}{6} = 17\%\ recall$

Precision

Candidate Text 1 : $\frac{3}{4} = 50\%\ precision$ Candidate Text 2 : $\frac{1}{6} = 17\%\ precison$

F1-Score

Candidate Text 1 : $\frac{3}{4} = 50\%\ f1 - score$ Candidate Text 2 : $\frac{1}{6} = 17\%\ f1 - score$

References

- https://towardsdatascience.com/the-ultimate-performance-metric-in-nlp-111df6c64460
- https://aclanthology.org/W04-1013.pdf