Project Report (Team - 6)

Title - Measure Text Fluency

Members:

- Aditya Rathi (2020201041)
- Prashant Raj (2020201057)
- Jayant Ingle (2020201019)

Measure Text Fluency

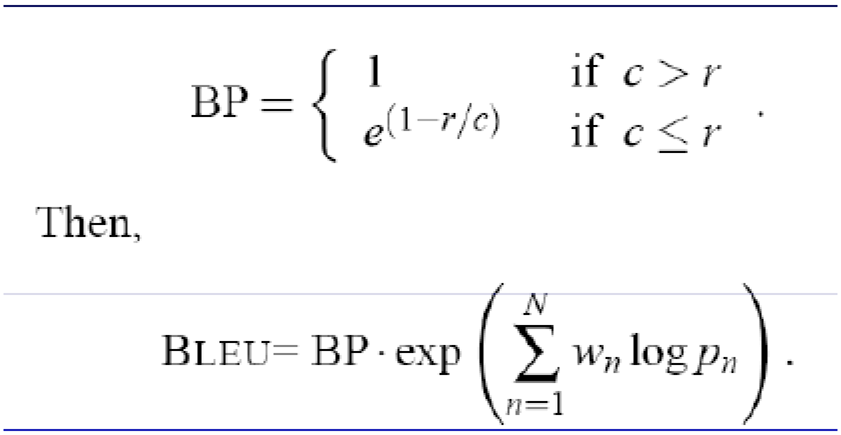
▼ Introduction

- The aim of this project is to implement metrics to evaluate text fluency of machine translations.
- Fluency is commonly considered as one of the dimensions of text quality of MT. Fluency measures the quality of the generated text, without taking the source sentence into account. It accounts for criteria such as grammar, spelling, choice of words, and style.
- Given a candidate sentence (a translation obtained from MT) and a reference translation, we try to implement different metrics to compute their fluency.

▼ Theory about different metrics

- **BLEU**
    - BLEU stands for Bi-Lingual Evaluation Understudy. It is a segment level algorithm that judges translations on a per-word basis.
    - BLEU measures MT adequacy by looking at word precision and MT fluency by calculating n-gram precision, returning a translation score on a scale from 0-1 (alternative: 0-100 scale). BLEU's n-gram matching requires exact word matches, meaning that if different vocabulary or phrases are used in reference translation, the score will be lower.

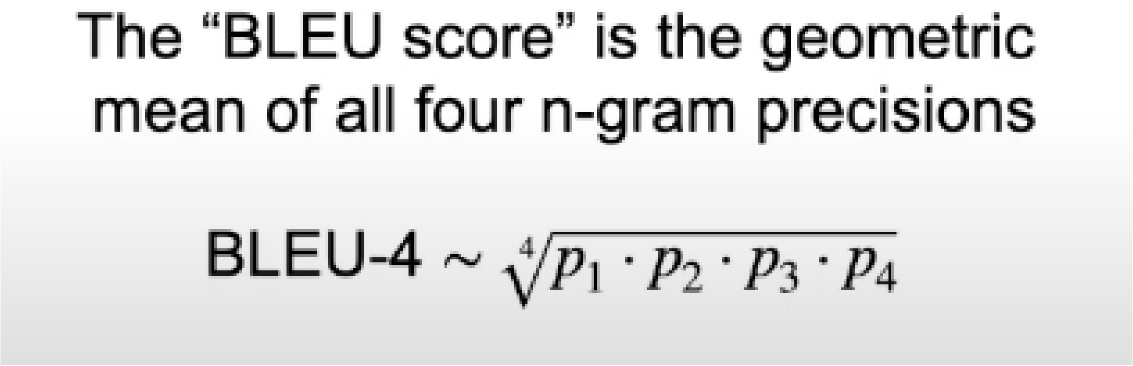
- BLEU computes the precision for several different N-Grams and then averages out the results.
- BLEU score is calculated by taking the geometric mean of all the precision scores.
- Below is the formula for computing BLEU score, where BP is brevity penalty, BLEU is the penalty multiplied with the geometric mean of all the precision scores for different n-gram values.



$$\mathrm{BP} = \begin{cases} 1 & \text{if } c > r \\ e^{(1-r/c)} & \text{if } c \le r \end{cases} .$$

Then,

$$\mathrm{BLEU} = \mathrm{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n \right) .$$



The “BLEU score” is the geometric mean of all four n-gram precisions

$$\text{BLEU-4} \sim \sqrt[4]{p_1 \cdot p_2 \cdot p_3 \cdot p_4}$$

- BLEU scores are generally better if there exists multiple human translations for each sentence.
- BLEU scores above 0.3 generally reflect the translations are understandable.
- BLEU scores above 0.5 reflect good and fluent translations.
- **Advantages -**

    - Fast and simple to calculate.
    - Widely used for MT evaluation.
    - Easy to implement.
  - **Disadvantages** -
    - Doesn’t incorporate sentence structures or morphology or synonyms.
    - Struggles with non-English language, specially with morphological rich languages..
    - Hard to compare scores with different tokenizers.
    - Not tunable to different target human measures or for different languages.
    - Mainly precision based, no concept of recall (compensates recall by using brevity penalty).
    - All words that are matched weigh equally in BLEU (loss of importance).
    - BLEU’s higher order n-grams account for fluency and grammaticality. Geometric n-gram averaging is volatile to “zero” scores.
    - As the selected translation for each segment may not be the only correct one, it is often possible to score good translations poorly.
    - The BLEU metric also gives higher scores to sequential matching words. That is, if a string of four words in the MT translation match the human reference translation in the same exact order, it will have more of a positive impact on the BLEU score than a string of two matching words will. This means that an accurate translation will receive a lower score if it uses different, but correct words or matching words in a different word order.

- **ROUGE**
  - ROUGE stands for Recall-Oriented Understudy for Gisting Evaluation. It is essentially a set of metrics for evaluating automatic summarization of texts as well as machine translations. It works by comparing an **automatically**

    produced summary or **translation** against a set of **reference summaries** (typically human-produced).

    - ROUGE is actually a set of metrics, rather than just one. We have used ROUGE-N for this project.
    - ROUGE-N measures the number of matching ‘n-grams’ between our model-generated text and a ‘reference’.
      An n-gram is simply a grouping of tokens/words. A unigram (1-gram) would consist of a single word. A bigram (2-gram) consists of two consecutive words:

      ```
      Original: "the quick brown fox jumps over"

      Unigrams: ['the', 'quick', 'brown', 'fox', 'jumps', 'over']

      Bigrams: ['the quick', 'quick brown', 'brown fox', 'fox jumps', 'jumps over']

      Trgrams: ['the quick brown', 'quick brown fox', 'brown fox jumps', 'fox jumps
      over']
      ```

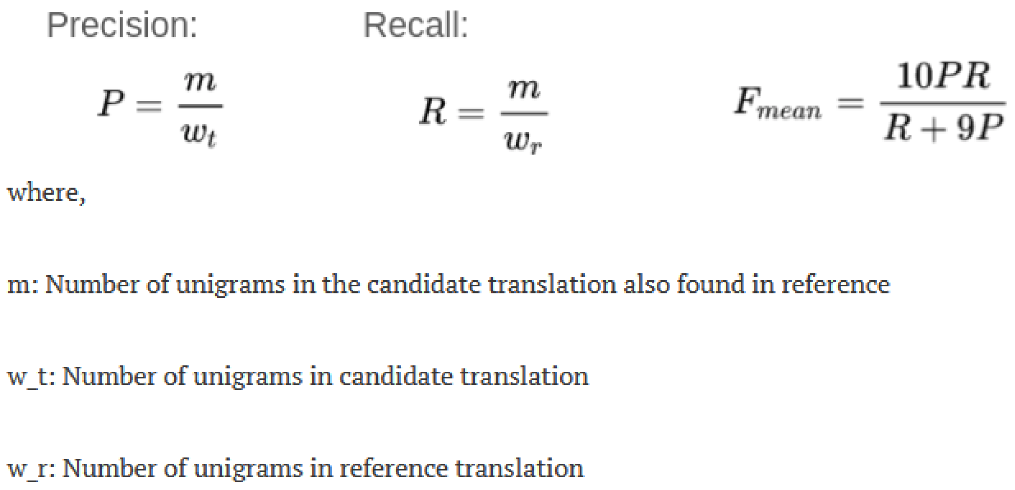
    - **Disadvantages** : ROUGE is a great evaluation metric but comes with some drawbacks. In-particular, ROUGE does not cater for different words that have the same meaning — as it measures syntactical matches rather than semantics.

      So, if we had two sequences that had the same meaning — but used different words to express that meaning — they could be assigned a low ROUGE score.

      This can be offset slightly by using several references and taking the average score, but this will not solve the problem entirely.

- **METEOR**
    - METEOR stands for Metric for Evaluation of Translation with Explicit Ordering.
    - Combine Recall and Precision as weighted score components, weighted towards Recall instead of using brevity penalty.

- Align MT output with each reference individually and take score of best pairing.
- Matching takes into account translation variability via word inflection variations, synonymy and paraphrasing matches.
- Addresses fluency via a direct penalty for word order rather than relying on higher order n-grams like BLEU.
- Parameters of metric components are tunable to maximize the score correlations with human judgements for each language.
- The parameters that can be tuned are Alpha, Beta and Gamma, where Alpha is importance given to precision vs recall, Beta controls the functional behaviour of word ordering penalty score, Gamma controls the relative importance of correct word ordering. These parameters can be tuned to maximize correlation of the scores with human judgements.
- Formula to compute METEOR score -



Precision:

$$P = \frac{m}{w_t}$$

Recall:

$$R = \frac{m}{w_r}$$

$$F_{mean} = \frac{10PR}{R + 9P}$$

where,

m: Number of unigrams in the candidate translation also found in reference

w_t: Number of unigrams in candidate translation

w_r: Number of unigrams in reference translation

$$p = 0.5\left(\frac{c}{u_m}\right)^3$$

C: Number of chunks in candidate

U_m: Unigrams in candidate

The final meteor score combines the F-score computed from precision and recall with the chunk penalty.

$$M = F_{mean}(1 - p)$$

  - Like BLEU, more reference translations do help in METEOR too but only marginally.
  - METEOR scores above 0.5 generally reflects understandable translations.
  - METEOR scores above 0.7 generally reflects good and fluent translations.
  - METEOR has been shown to consistently outperform BLEU in correlation with human judgements.
  - **Disadvantages**:
    - Is slower than BLEU and ROUGE.
    - Because METEOR computes word order, as the size of sentences increase, the number of possible combinations increase exponentially and hence complexity increases multi-fold.

- The aim of this project is to implement metrics to evaluate text fluency of machine translations.
- Fluency is commonly considered as one of the dimensions of text quality of MT. Fluency measures the quality of the generated text, without taking the source

sentence into account. It accounts for criteria such as grammar, spelling, choice of words, and style.

- Given a candidate sentence (a translation obtained from MT) and a reference translation, we try to implement different metrics to compute their fluency.

▼ Input and Output

- For sentence input.

```
NLP_Project$ time python3 main.py -t cmd -em all

Candidate sentence :I was a Ph.D. student in clinical psychology at Berkeley.
Reference sentence :I was the thesis student in clinical psychology in Berkeley.
bleu score : 0.32466791547509893
rouge score : 0.8333333333333334
meteor score : 0.6776

real  0m12.605s
user  0m1.271s
sys 0m0.921s
```

- For file input.

```
NLP_Project$ time python3 main.py -t file -em all -rf ../dataset/reference.en
-cf ../dataset/test.en -of output.txt

real  2m10.625s
user  2m10.520s
sys 0m1.630s
```

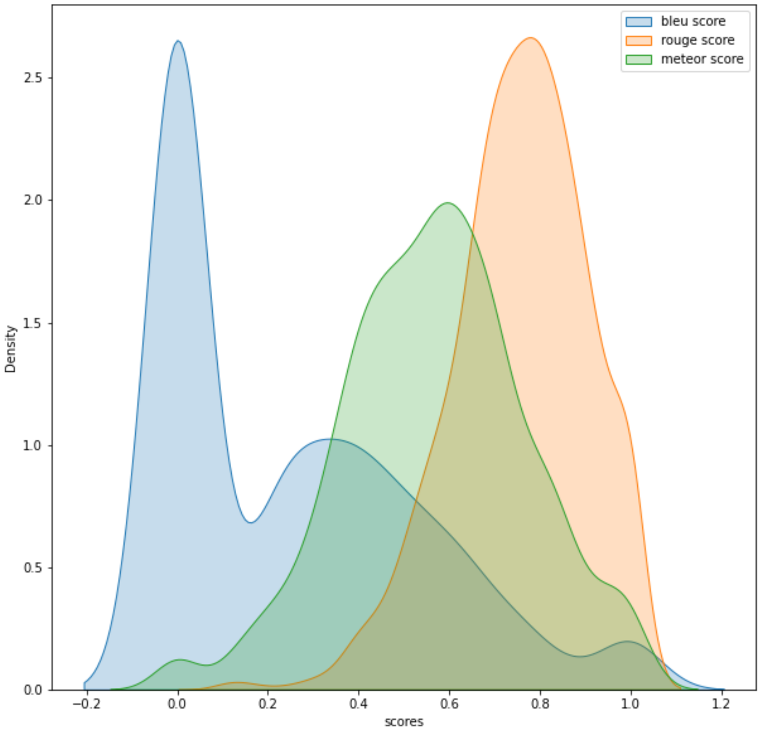
```
bleu score      rouge score     meteor score
3.908633169762327e-78 0.7169811320754716  0.40903192156521845
0.32466791547509893 0.8363636363636363  0.6776
8.416851712392762e-232  0.3684210526315789  0.11347071296050885
.
.
.
Average bleu score : 0.2526341112975136
Average rouge score : 0.7544767831340002
Average meteor score : 0.5671948244799403
Total bleu time : 0.469754695892334
Total rouge time : 0.08396792411804199
Total meteor time : 129.39375829696655
```

▼ Analysis, Results and Outcomes

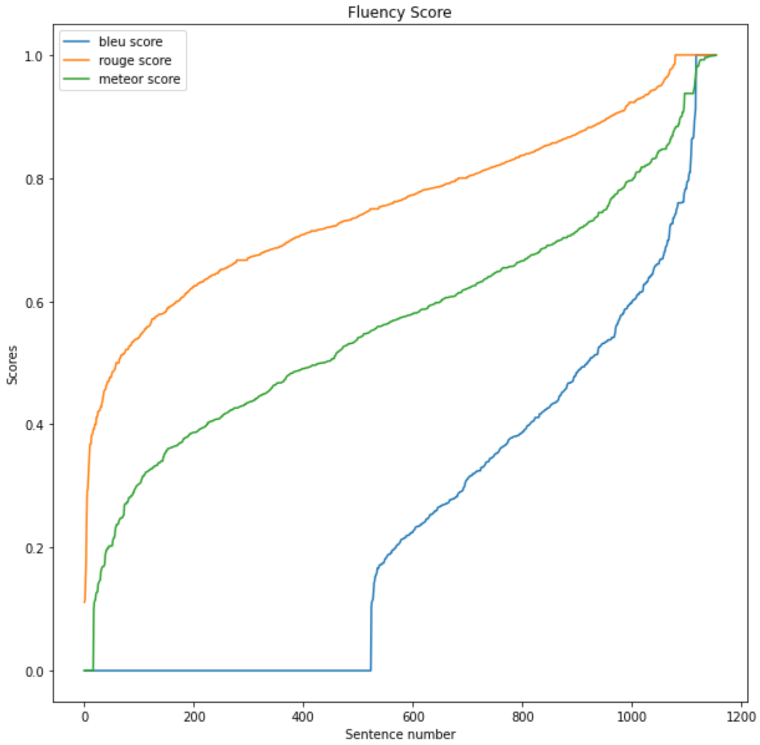
- Number of Sentences = 1154
- Average Scores

```
bleu score      0.208717
rouge score     0.763636
meteor score    0.572785
```

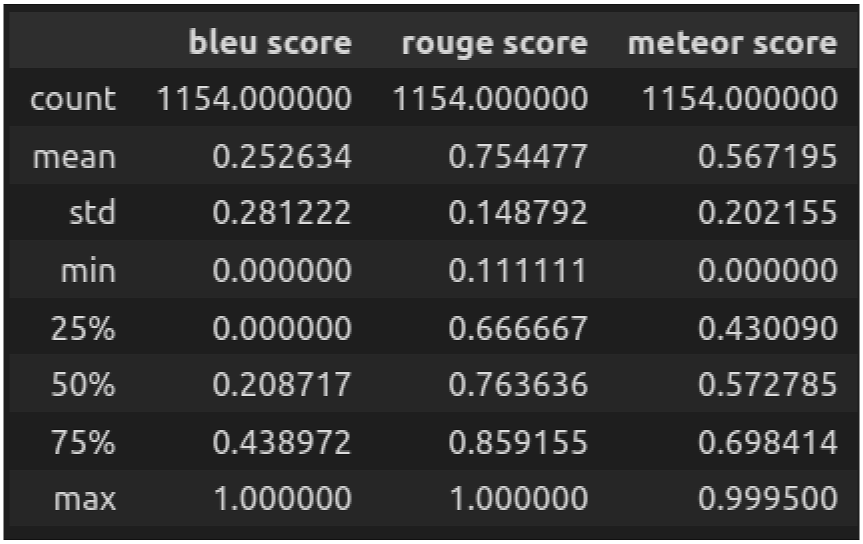
- Density Plot



- Sorted Fluency Score Plot



- Different Metrics



	bleu score	rouge score	meteor score
count	1154.000000	1154.000000	1154.000000
mean	0.252634	0.754477	0.567195
std	0.281222	0.148792	0.202155
min	0.000000	0.111111	0.000000
25%	0.000000	0.666667	0.430090
50%	0.208717	0.763636	0.572785
75%	0.438972	0.859155	0.698414
max	1.000000	1.000000	0.999500

- Average Time

```
Total bleu time : 0.5282821655273438
Total rouge time : 0.09468770027160645
Total meteor time : 151.73400831222534
```

▼ Conclusions

- When the sentence length increases, meteor takes more time to compute the score due to the computation of all possible combinations.
- Time comparison
  - METEOR Metric > BLEU Metric > ROUGE Metric
- Text Fluency comparison
  - ROUGE Metric > METEOR Metric > BLEU Metric
- BLEU is a precision score based metric, and doesn’t account for recall.
- Rouge and Meteor accounts for both precision and recall.
- Rouge also accounts for F1 score.

▼ References

- https://ieeexplore.ieee.org/document/1244655?arnumber=1244655
- https://aclanthology.org/K18-1031.pdf
- METEOR
  - https://www.youtube.com/watch?v=FqQbrlEh_b0
- BLEU
  - https://www.youtube.com/watch?v=M05L1DhFqcw
  - https://jaketae.github.io/study/bleu/
- ROUGE
  - https://www.youtube.com/watch?v=TMshhnrEXlg