Metrics for Automatic Evaluation of Text from NLP Models for Text to Scene Generation

S. Yashaswini and S. S. Shylaja

***Abstract* — Performance metrics give us an indication of which model is better for which task. Researchers attempt to apply machine learning and deep learning models to measure the performance of models through cost function or evaluation criteria like Mean square error (MSE) for regression, accuracy, and f1-score for classification tasks Whereas in NLP performance measurement is a complex due variation of ground truth and results obtained.**

***Kye words* — accuracy, f1-Score metrics, MSE, Regression.**

I. INTRODUCTION

In Natural Language Processing, there exists a bias in models due to the dataset or performance evaluation criteria. Hence there is a need to apply Standard Performance Benchmarks metrics to evaluate the performance of models for NLP tasks. NLP is widely used in the field of research for many applications like Machine translation, Question Answering, Text Summarization, Image captioning, Sentiment Analysis, etc. [1].

Automatic evaluation of natural language generation, for applications like machine translation and caption generation, requires comparing candidate sentences to annotated references. The goal is to evaluate semantic equivalence. The methods used will rely upon surface-form similarity. Generally, we evaluate the Machine-generated texts against a target text (truth value). The generated text refers to the machine-produced text output from the model and target or reference text refers to the original truth value text. The performance of subtask can be measured by applying Intrinsic Evaluation metrics which focus on intermediary objectives and Extrinsic Evaluation which focuses on the performance of the final objective. Carefully picking metrics is an important part of the ensuring system we work with becomes usable [2].

The Text generated from several NLP models coupled with ML techniques can be used to compare models in NLG Domain. The commonly used evaluation metrics are discussed below in Fig 1.

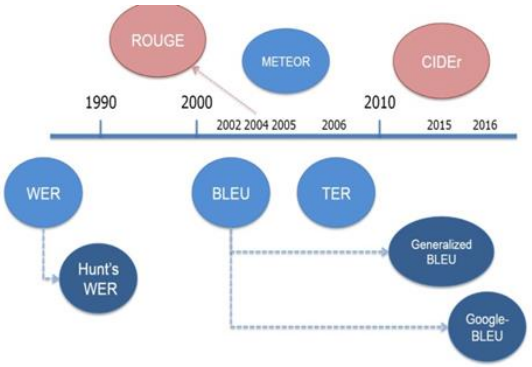

Fig. 1. Performance Evaluation Metrics.

II. IMPLEMENTATION

Natural language generation requires comparing candidate sentences to annotated references. Given reference set x with k labels has {x1, x2, x3, … Xn} and Candidate set y with l labels has {y1, y2, y3, …yn}. Evaluation metric $z = f(x,y) \in R$, the selection of evaluation metric depends on the type of NLP task or application choosing the better metric helps to provide correlation with human judgment. Existing metrics can be broadly categorized into using n-gram matching, edit distance, embedding matching [3].

The intrinsic metrics that are used to evaluate NLP systems are as follows:

Accuracy: The accuracy metric is used in classification tasks to learn the closeness of a measured value to a known value. It's typically used in instances where the output variable is categorical or discrete.

Precision: The precision metric would inform the number of labels that are labeled as positive in correspondence to the instances that the classifier labeled as positive.

Recall: Recall measures how well the model can recall the positive class. Recall value signifies the number of positive labels that the model has correctly identified as positive.

F1 Score: Precision and Recall are complementary metrics that have an inverse relationship. If both metrics are equally important then the F1 score can be used to combine precision and recall into a single metric [4].

The popular metrics available are built upon exact matching scores. The Metrics are listed below.

Bilingual Evaluation Understudy (BLEU): The BLEU score evaluates the quality of text that has been translated by a machine from one natural language to another. BLEU Score is a performance metric to measure the performance of

Submitted on June 30, 2021.
Published on, July 22, 2021.
S. Yashaswini, Cambridge Institute of Technology, India.
(e-mail: Yashaswini.cse@cambridge.edu.in)
S. S. Shylaja, PES University, India.
(e-mail: shylaja.sharath@pes.edu)

machine translation models. It evaluates how well a model translates from one language to another. The MT will compare on unigram, bigram, or trigram in output with ground truth. Some of its shortcomings of BLEU Scores are It doesn't consider meaning, sentence structure, and morphologically rich language [7].

	precision	recall	f1-score	support
B-LOC	0.810	0.784	0.797	1084
B-MISC	0.731	0.569	0.640	339
B-ORG	0.807	0.832	0.820	1400
B-PER	0.850	0.884	0.867	735
I-LOC	0.690	0.637	0.662	325
I-MISC	0.699	0.589	0.639	557
I-ORG	0.852	0.786	0.818	1104
I-PER	0.893	0.943	0.917	634
O	0.992	0.997	0.994	45355
accuracy			0.971	51533
macro avg	0.814	0.780	0.795	51533
weighted avg	0.970	0.971	0.971	51533

Fig. 2. Comparison of precision Vs. Recall vs. F1 – Score.

The BLUE score helps to evaluate the sentences related to interior design 76,068 sentences were considered as a reference set and 54 non repetitive sentences were taken as candidate set and precision of 47.74 and a BLEU Score of 46.59 was obtained as shown in Fig. 2.

METEOR: The Metric for Evaluation of Translation with Explicit ORdering (METEOR) is a precision-based metric for the evaluation of machine-translation output. It overcomes some of the pitfalls of the BLEU score, such as exact word matching whilst calculating precision. The METEOR score allows synonyms and stemmed words to be matched with a reference word.

The n-grams can be matched based on stemmed words and meanings. METEOR uses unigram precision and recall to compute a score.

ROUGE: Recall-Oriented Understudy for Gisting Evaluation (ROUGE) evaluation metric measures the recall. It's typically used for evaluating the quality of generated text and in machine translation tasks. However, since it measures recall it's mainly used in summarization tasks [5].

CHRF Score: character level n-grams play an important role in the automatic evaluation as a part of more complex metrics [8]. The n-gram based F-score; especially the linguistically motivated ones based on Part-of-Speech tags and morphemes correlate very well with human judgments outperforming the widely used metrics such as BLEU and TER.

NIST provides the evaluation infrastructure, where the source files being MT system output is used to assess the quality of the source files. The goal is to create correlation between metrics and human assessment. Different types of human assessment are used.

The plots given below helps us to understand the ROC based on performance metrics. The score ranges from the probabilistic value between 0 to 1. The values are scores of different metrics for 100 and 300 sentences as shown in Fig. 5.

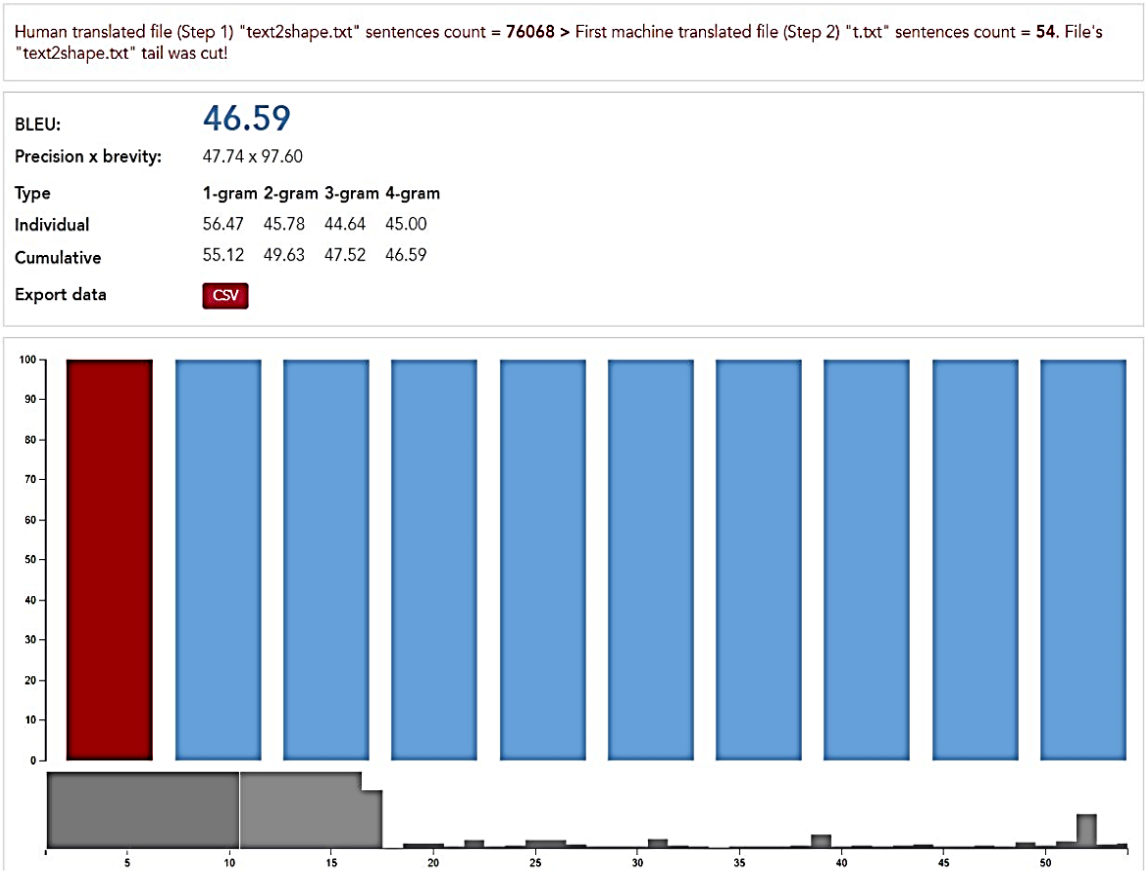

Fig. 3. The comparison of BLEU Score for reference sentences.

```
(base) C:\Users\yashaswini\Desktop\metrics>python main.py
{'Bleu_score': [0.8902579342581529], 'chrf_score': [0.8665575895977126], 'gleu_score': [0.891566265060241], 'Meteor_Scor
e': [0.0], 'nist_score': [0.1261922035977232], 'rouge_score': [0.9302328748062024]}
{'Bleu_score': [0.897752847848028], 'chrf_score': [0.8718223090938816], 'gleu_score': [0.898876404494382], 'Meteor_Score
': [0.0], 'nist_score': [0.12007743382732637], 'rouge_score': [0.9347829253623198]}
{'Bleu_score': [0.9000283069718913], 'chrf_score': [0.8747453425669116], 'gleu_score': [0.9010989010989011], 'Meteor_Sco
re': [0.0], 'nist_score': [0.11818274152505612], 'rouge_score': [0.936170529432625]}
dict_keys(['Bleu_score', 'chrf_score', 'gleu_score', 'Meteor_Score', 'nist_score', 'rouge_score'])
dict_values([[0.9000283069718913], [0.8747453425669116], [0.9010989010989011], [0.0], [0.11818274152505612], [0.93617052
9432625]])
{'Bleu_score': [0.9000283069718913], 'chrf_score': [0.8747453425669116], 'gleu_score': [0.9010989010989011], 'Meteor_Sco
re': [0.0], 'nist_score': [0.11818274152505612], 'rouge_score': [0.936170529432625]}
dict_keys(['Bleu_score', 'chrf_score', 'gleu_score', 'Meteor_Score', 'nist_score', 'rouge_score'])
dict_values([[0.9000283069718913], [0.8747453425669116], [0.9010989010989011], [0.0], [0.11818274152505612], [0.93617052
9432625]])
{'Bleu_score': [0.9000283069718913], 'chrf_score': [0.8747453425669116], 'gleu_score': [0.9010989010989011], 'Meteor_Sco
re': [0.0], 'nist_score': [0.11818274152505612], 'rouge_score': [0.936170529432625]}
dict_keys(['Bleu_score', 'chrf_score', 'gleu_score', 'Meteor_Score', 'nist_score', 'rouge_score'])
dict_values([[0.9000283069718913], [0.8747453425669116], [0.9010989010989011], [0.0], [0.11818274152505612], [0.93617052
9432625]])
{'Bleu_score': [0.9000283069718913], 'chrf_score': [0.8747453425669116], 'gleu_score': [0.9010989010989011], 'Meteor_Sco
re': [0.0], 'nist_score': [0.11818274152505612], 'rouge_score': [0.936170529432625]}

(base) C:\Users\yashaswini\Desktop\metrics>
```

Fig. 4. Comparison of scores for n (1, 10, 100, 1000) sentences.

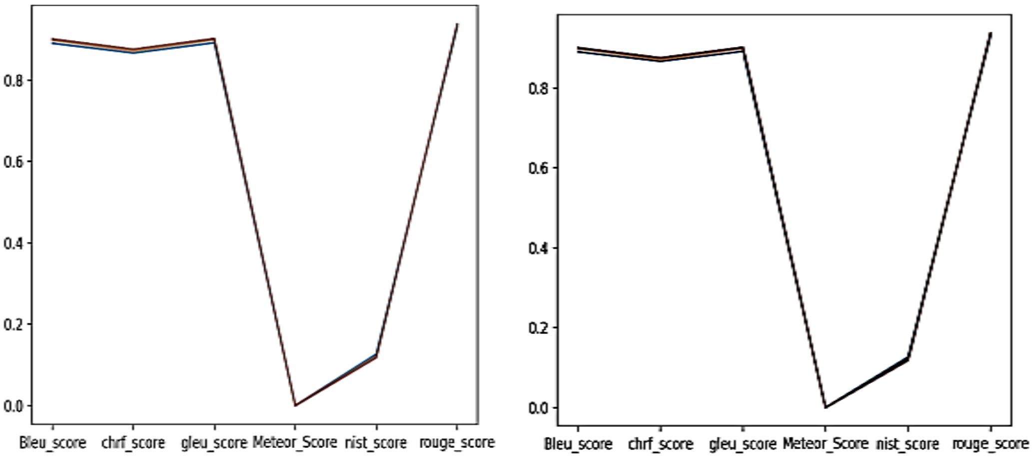

Fig. 5. Comparison graph between 100 and 300 sentences.

```
Model: "model_1"

Layer (type)                 Output Shape          Param #     Connected to
==================================================================================================
input_ids (InputLayer)       (None, 72)            0

input_masks (InputLayer)     (None, 72)            0

segment_ids (InputLayer)     (None, 72)            0

bert_layer_1 (BertLayer)     (None, None, 768)     110104890   input_ids[0][0]
                                                               input_masks[0][0]
                                                               segment_ids[0][0]

dense_1 (Dense)              (None, None, 18)      13842       bert_layer_1[0][0]
==================================================================================================
Total params: 110,118,732
Trainable params: 108,905,490
Non-trainable params: 1,213,242
```

Fig. 6. Bert model for POS Tagging.

BERT Score: BERT score leverages the pre-trained contextual embedding's from BERT and matches words in candidate and reference sentences by cosine similarity. It correlates human judgment with sentence-level evaluation. Moreover, BERT Score computes precision, recall, and F1 measure, which can be useful for evaluating different language generation tasks [12]. The accuracy of 97% is obtained for 900 sentences as shown in Fig 6.

BLEURT: It is an evaluation metric for Natural Language Generation. It is built using multiple phases of transfer learning starting from a pre-trained BERT model and then employing another pre-training phrase using synthetic data [6]. Finally, it is trained on human annotations [10]. You may run BLEURT out-of-the-box or fine-tune it for your specific application as shown in Fig. 8.

```
------------ Result of Bert fine-tuned model ----------

              precision    recall  f1-score   support

         ADJ     0.9136    0.8973    0.9054       224
         ADP     0.9659    0.9857    0.9757       488
         ADV     0.9587    0.8855    0.9206       131
         AUX     0.9957    1.0000    0.9979       234
       CCONJ     0.9896    0.9896    0.9896        96
         DET     0.9955    1.0000    0.9977       439
        INTJ     1.0000    1.0000    1.0000         2
        NOUN     0.9623    0.9841    0.9731       753
         NUM     0.9385    1.0000    0.9683        61
        PART     0.9565    1.0000    0.9778        66
        PRON     1.0000    0.9340    0.9659       106
       PROPN     0.9610    0.8222    0.8862        90
       PUNCT     1.0000    1.0000    1.0000       339
       SCONJ     0.9524    0.7843    0.8602        51
        VERB     0.9608    0.9785    0.9696       326
           X     1.0000    1.0000    1.0000         2

    accuracy                         0.9710      3408
   macro avg     0.9719    0.9538    0.9617      3408
weighted avg     0.9709    0.9710    0.9705      3408

Accuracy: 0.9710
f1-macro score: 0.9617
```

Fig. 7. Precision, Recall, and f1-Score.

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
Original tokens: ['there', 'is', 'a', 'room', 'with', 'chair', 'and', 'the', 'table', '.']
BERT tokens: ['[CLS]', 'there', 'is', 'a', 'room', 'with', 'chair', 'and', 'the', 'table', '.', '[SEP]']
Converting examples to features 100% 1/1 [00:00<00:00, 53.95it/s]

Word in BERT layer  | Initial word   : Predicted POS-tag
------------------------------------------------------------
there               | there          : ADV
is                  | is             : VERB
a                   | a              : DET
room                | room           : NOUN
with                | with           : ADP
chair               | chair          : NOUN
and                 | and            : CCONJ
the                 | the            : DET
table               | table          : NOUN
.                   | .              : PUNCT
```

Fig. 8. Pretrained Bert model with Predicted POS Tags.

III. Mathematical Formulations with Units

The most commonly used metrics for text generation is to count the number of n-grams that occur in the reference x and candidate y. formally, let S(x) and S(y) be the lists of token n-grams ($n \in Z+$) in the reference x and candidate y sentences. The number of matched n-grams is $P \in S(y)\ I[w \in S(x)]$, where I is an indicator function. The exact match precision (Exact-Pn) and recall (Exact-Rn) scores are:

$$\text{Exact-Pn} = P \in S(y)\ I[w \in S(x)\ ] / S(y) \quad \text{and}$$

$$\text{Exact-Rn} = P \in S(x)\ I[w \in S(y)\ ] / S(x)$$

The Units in the metrics are probability varying from 0 to 1. The value 0 indicating the least probability and value 1 indicates the highest probability.

IV. Results And Discussions

We evaluated metrics on considering the 1k sentences dataset got by applying the RNN-LSTM model for human-annotated sentences. The different automatic metrics like BLUE, CHRF, GLEU, METEOR, NIST, and ROGUE scores. The experimental setup uses 600 sentences as reference sentences and 300 has candidate sentences, a step-by-step evaluation is employed on 10, 100, 200, and 300 sentences, and comparative scores are noted down as shown in Table I.

The scores infer the BLEU score increases with an increase in the candidate sentences the results are based on bigram pairs. The CHRF score, GLEU score, and ROUGE[5] score have increased in the increase in no of candidate sentence whereas the NIST score has decreased and meteor score is zero since the sentence considered are interior design related hence adequacy and fluency error exists [11].

The experimentation results show that the Rouge score works well for interior design sentences by considering n-gram overlap scores. These scores are correlated with the human evaluation of summaries up to some level of accuracy. Nevertheless, Rouge scores are used to compare 2 candidate summarization systems as shown in Fig. 8. The best evaluation policy is collecting human judgments provided there is sufficient time and cost. Recently scoring criteria are used in summarization tasks.

TABLE I: PERFORMANCE METRIC SCORES ON N (10,100,200,300) CANDIDATE SENTENCES

Corpus Length	BLEU Score	ChrF Score	GLEU Score	METEOR Score	NIST Score	ROUGE Score	Remarks
1	0.89	0.866	0.891	0	0.126	0.93	Rouge↑
10	0.897	0.871	0.898	0	0.12	0.934	Nist ↓
100	0.9	0.874	0.901	0	0.118	0.936	Nist ↓
200	0.9	0.874	0.901	0	0.118	0.936	---
300	0.9	0.874	0.901	0	0.118	0.936	---

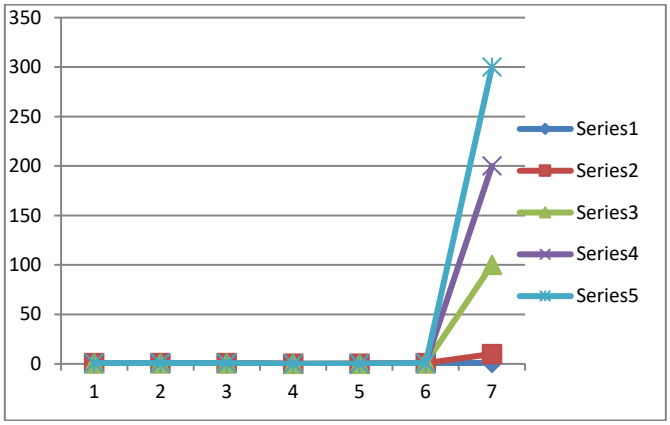

Fig. 8. Rogue Score for dataset.

The experiments were further carried out for n iterations to check whether the values are consistent for n batches in the dataset. The scores remained approximately the same for iterations hence it's determined the rogue score helps with matching n-grams in all batches of reference and candidate summaries as shown in Fig. 9.

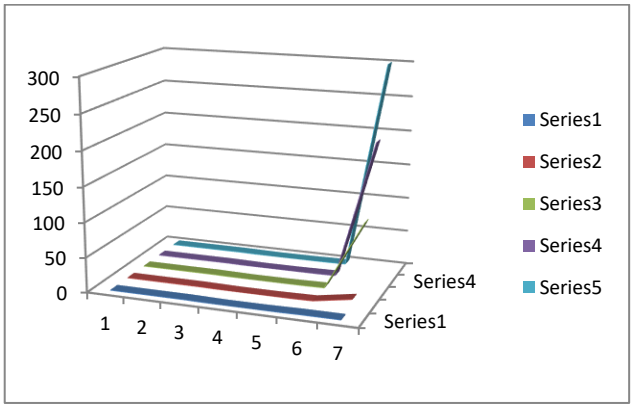

Fig. 9. Metrics values after executing for n trials.

V. CONCLUSION

The experimentation carried out shows that the rogue scores outperform by considering the n overlap n-grams. The meteor scores are zero irrespective of n iterations due to the dataset lacking fluency and adequacy. The interior design dataset is generated by applying the rnn-lstm model to human-annotated sentences. The dataset is generated and not translated hence it is hard to obtain fluency and adequacy.

REFERENCES

[1] Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In Proceedings of the ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization, pages 65–72.

[2] George Doddington. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In Proceedings of the second international conference on Human Language Technology Research, pages 138–145.

[3] Morgan Kaufmann Publishers Inc. Hideki Isozaki, Tsutomu Hirao, Kevin Duh, Katsuhito Sudoh, and Hajime Tsukada. 2010. Automatic evaluation of translation quality for distant language pairs. In Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, pages 944–952. Association for Computational Linguistics.

[4] Guillaume Klein, Yoon Kim, Yuntian Deng, Vincent Nguyen, Jean Senellart, and Alexander M. Rush. 2018. Opennmt: Neural machine translation toolkit.

[5] Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. Text Summarization Branches Out.

[6] Nitin Madnani. 2011. ibleu: Interactively debugging and scoring statistical machine translation systems. In 2011 IEEE Fifth International Conference on Semantic Computing pages 213–214. IEEE.

[7] Kishore Papineni, Salim Roukos, Todd Ward, and WeiJing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In Proceedings of the 40th annual meeting on association for computational linguistics, pages 311–318. Association for Computational Linguistics.

[8] Maja Popovic. 2015. chrf: character n-gram f-score ´ for automatic mt evaluation. In Proceedings of the Tenth Workshop on Statistical Machine Translation, pages 392–395.

[9] Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In Proceedings of the association for machine translation in the Americas, volume 200.

[10] Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. arXiv preprint arXiv:1609.08144.

[11] Qi Ye, Sachan Devendra, Felix Matthieu, Padmanabhan Sarguna, and Neubig Graham. 2018. When and why are pre-trained word embeddings useful for neural machine translation. In HLT-NAACL.

[12] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. arXiv preprint arXiv:1904.09675.

Yashaswini S. presently is working as Assistant Professor in Department of CSE, Cambridge Institute of Technology, Bangalore, she is Having 8 years of Teaching Experience and almost 2 years of industrial experience She is Currently pursuing her PhD in text to scene generation under VTU Belagavi.
She has published many research papers which includes 6 international journal, 3international conference and 4 national conference papers, she has proactively Participated in many FDP'S and workshop, as a part of industrial interaction has participated FDP on "Business Process Modeling Tools and Process Model" at J.P. Morgan PVT LTD and "Cyber Forensic workshop at NCS" at Nihon Communications PVT LTD. Her Areas of Interests include Digital Image Processing, NLP, Machine Learning, Computer Networks, cloud computing, and Information Security.

Dr. Shylaja S. S. is the Chairperson, Department of Computer Science and Engineering (UG Studies) at P E S University, Bangalore. She is also the Professor and Head, Department of CSE and Department of ISE at P E S Institute of Technology (PESIT), Bangalore. She completed her Bachelor's degree from UVCE, Bangalore University in 1989, Master's degree from Sri Jayachamarajendra College of Engineering, Mysore University in 1993.

She secured the First rank in the university in her M. Tech course. She completed the 'C' level course from DOEACC and holds a Ph.D. in the domain of Face Recognition from Visvesvaraya Technological University (VTU). She has 28 years of teaching experience and 15 years of research experience.

She has several journal articles, national and international conference publications to her credit. She has taught a plethora of courses and has guided several projects and research activities at undergraduate and postgraduate levels. She is a member of BOS of several organizations, reviewer of conferences and member of many technical committees.

Dr. Shylaja is heading two centers at P E S University, the Center for Data Sciences and Applied Machine Learning (CDSAML) and the Huawei Innovation Lab, facilitating research and internship opportunities for students and faculty members. Her research areas include Image Processing, Computer Vision, Natural Language Processing and Machine Learning.