LSTM and GRU neural network performance comparison study

Rohit Patil

*Department of Computer Science and Business System*

*JSPM’s Rajarshi Shahu College of Engineering,Tathawade*

Pune,India

[rohitspatil30@gmail.com](mailto:rohitspatil30@gmail.com)

***Abstract*—*This paper briefs about the different transformer architectures - T5, BART, and Pegasus and their functioning. Finally, a comparative analysis of these models when used on the same data is presented. The summaries generated by these models are compared to a manually generated summary and ROUGE1, ROUGE2, and ROUGEL values are weighed. The purpose of this review on abstractive text summarization is to render a complete understanding of the elements of recent abstractive text summarization models as well as to provide an instinct of the challenges with these systems.***

***Keywords: Abstractive text summarization, Natural Language Processing, ROUGE, Transformers.***

1. BASICS
2. *TEXT SUMMERISATION:*

There is a huge quantity of data which is growing every day. This data is unorganized, there is an acute need for this data’s size to be reduced and summarized in a succinct manner. The purpose of automatically producing text summaries is to have summaries that are on par with human written documents. Data reduction alone is insufficient, the generated summaries must be precise and consistent. In 2016, Text summarization using the seq2seq model outperformed other models and demonstrated state-of-the-art performance amongst other models, where an attentional encoder-decoder RNN[3](Graves, 2013), which was actually established for machine translation, outperformed other models and demonstrated state-of-the-art performance among other models developed at the time.

In the realm of text summarization, more models were developed which aided in the creation of a more abstractive summarised result. One such model’s foundation is the standard feed-forward Network Neural Language Model (NNLM) which is used to estimate the contextual probability of the following word, also known as the next word prediction model. With the introduction of BERT, there was a broad range of progress in NLP tasks. BERT introduced pretrained language models that perform as a State-Of-The-Art model in NLP applications using a transfer learning approach. With all of its transfer learning features, BERT has left a lasting impression in the text summarization field. Another recent method of abstractive summarization is PEGASUS[1](Zhang, 2019), which combines gap sentence generation (GSG) and masked language model (MLM) to achieve a state-of-the-art result with a lesser sample size. T5 (Text-to-Text-Transfer-Transformer), a recent Google release, claims to surpass existing high-end algorithms such as BERT, GPT2, and others on NLP tasks like text classification, question answering, text summarization, etc.

1. *ENCODER DECODER ARCHITECTURE:*

The selection of encoder-decoder architecture provides us with certain choices of designing our encoder and decoder with standard RNN/ LSTM/ GRU, bidirectional RNN/LSTM/GRU, Transformer, BERT/GPT-2 architecture, or the very recent BART model.

1. *TRANSFORMERS:*

Transformers were a breakthrough introduced by Google for sequence learning tasks. Transformers are based entirely on attention mechanisms thus eliminating the requirement for recurrent as well as convolutional units. The transformer architecture consists of encoders and decoders stacked up.

The encoder and decoder blocks are made up of attention units and feed-forward units. The encoder part is a stack of six encoder units and the decoder part is a stack of six identical decoder units. Each encoder unit has a multi-head attention unit as well as a feedforward unit. Each decoder unit has an additional masked multi-head attention unit in addition to the feedforward unit and the multi-head attention unit.

The functioning of the transformer starts with the word embeddings of the input sequence. The word embeddings are forwarded to the first encoder which is then transformed and passed on to the following encoder. This is repeated many times until it gives the output.

1. MODEL TRAINING
2. *DATASET*

From here on out, the basic experimental setup is outlined, assessment metrics, and numerous models are analysed. Apart from this, findings from the research will be compared with the models’ performance. The dataset is derived from a text categorization dataset, which consists of BBC news website documents referring to articles featured in the paper

1. *TRANSFORMER ARCHITECTURES USED*
2. *Bidirectional and Autoregressive Transformers (BART)*

BART comprises two major components, a bidirectional encoder and a decoder. It is quite similar to BERT but pretrained on "facebook/bart-large-cnn" and then uses a tokenizer. This tokenizer is based on the GPT-2 tokenizer. The encoder is fairly similar to BERT and the decoder similar to GPT-2. The decoder used in BART is autoregressive in nature and this when regulated can be used for text summarization(NLP task). It uses denoising as the pre-training purpose. 6 layers in each, the encoder and the decoder are used in the base model of BART, whereas this number becomes 12 when it comes to the large model. Fine-tuning BART is helpful in applications such as sequence classification, token classification, sequence generation, and machine translation as the representations produced by it are extensively used by these applications.[7](Lewis, 2019)

1. *T5 (Text-to-text transfer transformer)*

T5 was trained on a huge amount of text in transfer learning before fine tuning on a downstream task. Seq-to-seq technique is used. Through crossed-attention layers, the encoded input is transmitted on decoder. Output generated by the decoder is of autoregressive nature. A sequence of tokens is given to the encoder to be mapped to a series of embeddings.[9](Raffel, 2019) The encoder is made up of two parts: a self-attention layer and a feed forward network. Before proceeding to each selfattention layer, there is a general attention mechanism which differentiates encoder from decoder; or else, their structures are alike. As a result, previously developed outputs can be utilised. The decoder's output is then transferred to a second dense layer, with softmax as the activation function.The input embedding matrix consists of weights from this layer’s outputs.

1. *Pegasus*

PEGASUS, developed by Google, expands to Pre-training with Extracted Gap-sentences for Abstractive Summarization Sequence-to-sequence models. The most important lines from the input data are extracted and they are then compiled as separate outputs. Also, choosing the most relevant sentences is better than randomly selecting sentences.[11](Zhang, 2019) This model is one of the most preferable models for abstractive summarization because it is similar to the ways humans generate a summary by reading the entire document and then producing a summary. The model is pre-trained on the newspaper CNN/DailyMail datasets.

1. RESULT ANALYSIS

**Quantitative Analysis**: The Rouge scores for all three models are compared and the models have used the same data while doing so. The results state that Pegasus has performed better than the other models.

|  |  |  |  |
| --- | --- | --- | --- |
| MODELS | ROUGE-1 | ROUGE-2 | ROUGE-L |
| T5 | 0.327122 | 0. 087318 | 0.173913 |
| BART | 0.245524 | 0.066838 | 0.143223 |
| PEGASUS | 0.351351 | 0.217391 | 0.243243 |

T5 (Text-to-Text-Transfer-Transformer), a recent Google release, claims to surpass existing high-end algorithms such as BERT[2](Devlin, 2018), GPT2, and others on NLP tasks like question answering, text classification, text summarization etc. Another transformer model by Facebook is BART. The latest released model for abstractive summarization is PEGASUS, which combines masked language model (MLM) and gap sentence generation (GSG) to achieve a state-of-theart result with a lesser sample size. All these three models were used to generate abstractive text summaries on the same input text.

The metric used for evaluation of the summaries generated is ROUGE(Recall-Oriented Understudy for Gisting Evaluation) score. The various system generated summaries are compared to manual summaries often known as the reference summaries. ROUGE-1 is the measure of the number of overlaps of unigrams in the system generated summary and the reference summary. ROUGE-2 is the measure of the number of overlaps of bigrams in the system generated summary when compared to the reference summary. ROUGE-L is the measure of a sequence of words that is common to both reference and system generated summary and is longest possible. The matches may or may not be successive matches. On comparative analysis of these 3 models- T5, BART and Pegasus, it was found that Pegasus outperforms the other two models and has higher rouge scores. It also generates better summaries for large input texts.

1. CONCLUSION

The pre-trained models, which were based on the transformer architecture, were executed for the purpose of summarization. The conclusion drawn from this analysis was that finely tuned transformers gave good results. The ROUGE scores[8](Lin, 2004) were computed for the summaries generated by each of the models and weighed against each other for precision, recall, and f-measure. The findings suggest that Pegasus gave results that outperformed the other two models. Future Scope of this research could be to implement a crossover of these models to improve text summarization in terms of accuracy and coherence of the outlines.

V. References

1. Allahyari M, Pouriyeh SA, Assefi M, Safaei S, Trippe ED, Gutierrez JB, Kochut K (2017) ‘Text Summarization Techniques: A Brief Survey’, CoRR abs/1707.02268:.
2. Devlin, J., Chang, M.-W., Lee, K. and Toutanova, K. (2018) ‘BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding’, Available at: https://arxiv.org/abs/1810.04805.
3. Graves A (2013) ‘Generating Sequences With Recurrent Neural Networks’, CoRR abs/1308.0850
4. Greene D, Cunningham P (2006) ‘Practical Solutions to the Problem of Diagonal Dominance in Kernel Document Clustering’, In Proceedings of the 23rd International Conference on Machine Learning Association for Computing Machinery, New York, NY, USA, pp 377–384 , October 2020.
5. Hochreiter S, Schmidhuber J (1997) ‘Long Short-Term Memory’ available https://doi.org/10.1162/neco.1997.9.8.1735
6. Kalchbrenner, N. and Blunsom P. (2013) ‘Recurrent Continuous Translation Models’, Association for Computational Linguistics, pp.1700–1709. Available at: <https://aclanthology.org/D13-1176.pdf>, March 2018.
7. Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V. and Zettlemoyer, L. (2019) ‘BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension’ Available https://arxiv.org/abs/1910.13461 [Accessed 9 Nov. 2021]. October 2018.
8. Zhang, J., Zhao, Y., Saleh, M. and Liu, P.J., (2019) ‘PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization’ 17 March 2020.