Research Proposal

Nirbhay P. Tandon
MSc. Data Science
Research Title:
Compare & Contrast Existing Transformer Architectures To Develop A New Architecture

Abstract

Attention-based Transformer architectures have become the norm of current Natural Language Processing applications. Google began this trend back in 2017 with their paper *Attention Is All You Need*[1], by introducing the Transformer architecture that works solely on attention mechanisms. The purpose of our work will be to explore a new kind of Transformer architecture. Compare & contrast its performance against other architectures such as BERT[4], RoBERTa[5], etc. that are also based on SQuAD 2.0 Dataset[2]. Through our research, we aim to produce a new Transformer Architecture that has a better performance than existing models for Question Answering in a conversational manner.

Contents

1 Introduction

The area of Natural language Processing has taken significant leaps in the last two decades. Work was done towards improving the ability of machine learning models to first recognize words, then sentences, followed by contextual understanding has led to several interesting & novel approaches in the field. From early on neural networks to creating Long Short-Term Memory architectures[29] by Sepp Hochreiter & Jurgen Schmidhuber helped in the mid-'90s that resolved the vanishing gradient problem of classical neural networks, we have come a long way.

The latest advancements in this field come from Google's research lab in the form of *Transformers*[1]. In their paper Attention Is All You Need[1] Vasvani et. al demonstrated how replacing the *encoder-decoder* based recurrent layers with *multi-headed self-attention* based ones removes the need for recurrence and convolutions entirely.

We have divided our research proposal into 7 sections. In Section 1 we shall take a look at briefly introducing the concept & why our work is necessary. Next, in Section 2, we outline the background work that has already been done in this field and how some of the papers relate to the work that has been done. We use this as an opportunity to highlight some of the shortcomings in current architectures & modelling techniques. In Section 3 we briefly outline the aim of our proposed research. In Section 4 we define in some detail the work that we will do to establish our research & how we plan to quantify the work that shall be done. In Section 5 we have highlighted that our goal is to produce a new transformer architecture that performs better at Question Answering based tasks & is capable of doing so in a conversational manner. In Section 6 we have outlined the minimum hardware requirements along with the resources available to the author. Finally, in Section 7 we submit a Gantt Chart to outline our plan against the number of weeks.

2 Background & Related Research

In this section, we shall highlight why this research has been conducted, what has led us this far & some of the interesting challenges that it poses. In 2.1, we briefly look at the history of Natural Language Processing & how some of the challenges were addressed. In 2.2, we look at the most interesting & latest research that has gone into creating the Transformer architecture, identify some of the common patterns & use that information

to strategise our model in other sections.

2.1 Background

2.2 Related Research

Outlined below are some of the most important pieces of research in the field of Natural Language Processing based neural networks, long short-term memory architectures & transformers.

1. Work done in the field of Long short-term & gated recurrent [7, 10] neural networks in particular, has been established as state of the art approach in sequence modelling, transduction problems such as language modelling and machine translation. In their paper Attention Is All You Need [1], Vasvani et. al set out to resolve problems in the parallelization & increased compute times of recurrent models. The inherently sequential nature highlights the issues in memory constraints, leading to reduced batch sizes. The architecture for a *Transformer* in this paper is outlined as having an encoder that maps input sequences to a continuous representation. This is then decoded into an output sequence of symbols one at a time. Each step is auto-regressive, i.e. it consumes the previously generated symbols as additional input when creating the next. This is similar to an ensemble tree architecture. Stacks of 6 encoder layers & 6 decoder layers is used.
   The encoder layers each have 2 sub-layers of a multi-head self-attention & the other a simple, position-wise fully connected feed-forward network layer.
   The decoder layer is similar to the encoder layer & has an additional 3rd sub-layer that performs multi-head attention over the output of the encoders. There is also normalization & the outputs are prevented from attending to subsequent positions.
   The attention mechanism can be described as mapping a query to a set of key-value pairs[1]. The evaluations performed on the Wall Street Journal dataset[3], using 40k sentences, showed that even without task-specific tuning the model had better results.

2. [2]

3. [4]

4. [5]

5. [20]

6. [16]
7. [14]
8. [13]
9. [18]

3 Aims & Objectives

The main aim of this research is to propose a new transformer architecture that can perform better at conversational Question & Answering from the SQuAD 2.0 dataset[2]. We shall:

1. Implement the existing models that are available via libraries such as HuggingFace[6], PyTorch & Tensorflow, run the SQuAD 2.0[2]
2. Obtain F1, validation, etc. scores for existing models & treat them as our benchmark scores
3. Identify drawbacks of the current architectures
4. Design our architecture & evaluate its performance
5. Fine-tune the architecture, re-evaluate & report improvements
6. Compare the results of our Transformer model with the benchmark scores.

We hope to establish our proposed transformer architecture as a competent enough contender to be used within both industry & academia.

4 Research Methodology

To implement this research we shall break the project down into 5 phases. These are outlined below.

4.1 Literature Review

In this phase, we will

1. Review the research that has been published already around the different kinds of architectures
2. Shortlist some of the most widely used ones, compare their results using the SQuAD 2.0 [2] & outline the pros and cons of each of these architectures.

The rationale is to review & understand as much of the previously established research as possible so that we can avoid potential pitfalls, not duplicate our efforts by reinventing the wheel & organize a better approach to perform our research. A lot of the background work available has already been highlighted in

4.2 Research Benchmarks

Here we shall focus on obtaining benchmark scores for the shortlisted architectures above using the dataset[2]. The parameters used will be validation & test set scores of the models. The training shall be carried out on each of the models for 100 epochs. We shall also look at the specificity/recall of these results to better understand if our work was done correctly or not.

4.3 Architecture Creation

In this section we will:

1. Mathematically model a new transformer architecture
2. Code the architecture
3. Run sample dataset to identify base benchmarks
4. Run the SQuAD 2.0 dataset[2] to obtain 1st pass performance benchmarks
5. Document architecture performance, identify pros & cons

4.4 Architecture Refinement

In this phase, we will focus on:

1. Reviewing the results from the previous section
2. Identifying the areas of improvement
3. Hypothesise the improvements & implement them in the architecture
4. Run the SQuAD 2.0 dataset[2] to obtain new performance benchmarks
5. Document architecture performance, identify pros & cons

4.5 Research Findings

In this subsection, we shall highlight the achievements of our research & draw comparisons to the benchmarks we obtained in 4.2.

4.6 Conclusion

Finally, here we aim to successfully conclude that our model performs better than existing models.

5 Expected Outcomes

We expect that our created model is at-par, if not better performing than existing models for Question Answering based problems.

6 Requirements & Resources

To successfully deliver on our research we will be utilizing the following hardware:

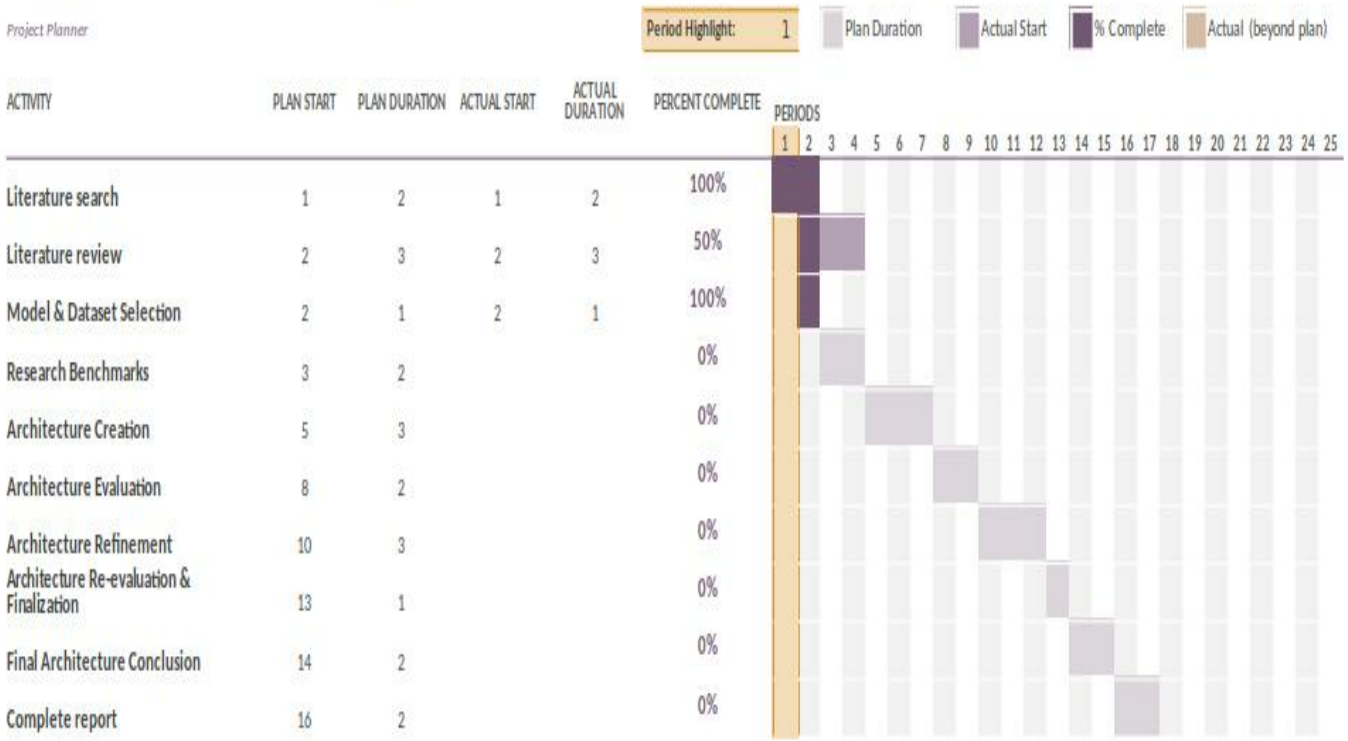
- EVGA GeForce RTX 2070 SUPER KO GAMING, 08G-P4-2072-KR, 8GB GDDR6, Dual Fans[25]. This graphics card is based on the Nvidia "Turing" architecture & has 2560 CuDA cores.
- Intel 10700 processor. 8 cores, 16 threads, 16M cache[26].
- VENGEANCE® LPX 8GB (1 x 8GB) DDR4 DRAM 2400MHz C14 Memory Kit - Black[27]. 8GB x 4, 32 GB total.
- Minimum 650W power supply.
- Minimum B550 Motherboard architecture. We will use the Asus Prime B550 motherboard[28].

The above hardware is available to the author & any changes to the same will be notified/highlighted in the subsequent reports.

7 Research Plan

Shown on the next page is the Gantt Chart highlighting the research stages & timelines.

Compare & Contrast Existing Transformer Architectures To Develop A New Architecture



References

[1] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, Aidan N, Kaiser, L. and Polosukhin, I. (2017). Attention Is All You Need. [online] arXiv.org. Available at: https://arxiv.org/abs/1706.03762.

[2] Rajpurkar, P., Jia, R. and Liang, P., 2018. Know what you don't know: Unanswerable questions for SQuAD. arXiv preprint arXiv:1806.03822.

[3] Marcus, M., Santorini, B. and Marcinkiewicz, M.A., 1993. Building a large annotated corpus of English: The Penn Treebank.

[4] Devlin, J., Chang, M.W., Lee, K. and Toutanova, K., 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

[5] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L. and Stoyanov, V., 2019. Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692.

[6] Huggingface.co. 2021. Transformers — transformers 4.4.2 documentation. [online] Available at: ¡https://huggingface.co/transformers/¿ [Accessed 4 April 2021].

[7] Hochreiter, S., Bengio, Y., Frasconi, P. and Schmidhuber, J., 2001. Gradient flow in recurrent nets: the difficulty of learning long-term dependencies.

[8] Akbik, A., Blythe, D. and Vollgraf, R., 2018, August. Contextual string embeddings for sequence labeling. In Proceedings of the 27th international conference on computational linguistics (pp. 1638-1649).

[9] Al-Rfou, R., Choe, D., Constant, N., Guo, M. and Jones, L., 2019, July. Character-level language modeling with deeper self-attention. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 33, No. 01, pp. 3159-3166).

[10] Zhou, J., Cao, Y., Wang, X., Li, P. and Xu, W., 2016. Deep recurrent models with fast-forward connections for neural machine translation. Transactions of the Association for Computational Linguistics, 4, pp.371-383.

[11] Luong, M.T., Le, Q.V., Sutskever, I., Vinyals, O. and Kaiser, L., 2015. Multi-task sequence to sequence learning. arXiv preprint arXiv:1511.06114.

[12] Bahdanau, D., Cho, K. and Bengio, Y., 2014. Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.

[13] Zhang, Y., Zhong, V., Chen, D., Angeli, G. and Manning, C.D., 2017, September. Position-aware attention and supervised data improve slot filling. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (pp. 35-45).

[14] Levy, O., Seo, M., Choi, E. and Zettlemoyer, L., 2017. Zero-shot relation extraction via reading comprehension. arXiv preprint arXiv:1706.04115.

[15] Yih, S.W.T., Chang, M.W., Meek, C. and Pastusiak, A., 2013. Question answering using enhanced lexical semantic models.

[16] Schmidhuber, J., 2015. Deep learning in neural networks: An overview. Neural networks, 61, pp.85-117.

[17] Chung, M.I., Kushner, W. and Damoulakis, J., 1985, April. Word boundary detection and speech recognition of noisy speech by means of iterative noise cancellation techniques. In ICASSP'85. IEEE International Conference on Acoustics, Speech, and Signal Processing (Vol. 10, pp. 1838-1838). IEEE.

[18] Peters, M.E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K. and Zettlemoyer, L., 2018. Deep contextualized word representations. arXiv preprint arXiv:1802.05365.

[19] Zhang, Z., Yang, J. and Zhao, H., 2020. Retrospective reader for machine reading comprehension. arXiv preprint arXiv:2001.09694.

[20] Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P. and Soricut, R., 2019. Albert: A lite bert for self-supervised learning of language representations. arXiv preprint arXiv:1909.11942.

[21] Graves, A., 2013. Generating sequences with recurrent neural networks. arXiv preprint arXiv:1308.0850.

[22] Weissenborn, D., Wiese, G. and Seiffe, L., 2017. Making neural qa as simple as possible but not simpler. arXiv preprint arXiv:1703.04816.

[23] Baevski, A. and Auli, M., 2018. Adaptive input representations for neural language modeling. arXiv preprint arXiv:1809.10853.

[24] Zhang, Z., Wu, Y., Zhou, J., Duan, S., Zhao, H. and Wang, R., 2020, April. Sg-net: Syntax-guided machine reading comprehension. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 34, No. 05, pp. 9636-9643).

[25] EVGA. 2021. EVGA - EU - Products - EVGA GeForce RTX 2070 SUPER KO GAMING, 08G-P4-2072-KR, 8GB GDDR6, Dual Fans - 08G-P4-2072-KR. [online] Available at: ¡https://eu.evga.com/products/product.aspx?pn=08G-P4-2072-KR¿ [Accessed 16 April 2021].

[26] Ark.intel.com. 2021. Intel® Core™ i7-10700 Processor (16M Cache, up to 4.80 GHz) Product Specifications. [online] Available at: ¡https://ark.intel.com/content/www/us/en/ark/products/199316/intel-core-i7-10700-processor-16m-cache-up-to-4-80-ghz.html¿ [Accessed 16 April 2021].

[27] Corsair.com. 2021. VENGEANCE® LPX 8GB (1 x 8GB) DDR4 DRAM 2400MHz C14 Memory Kit - Black. [online] Available at: ¡https://www.corsair.com/uk/en/Categories/Products/Memory/VENGEANCE-LPX/p/CMK8GX4M1A2400C14¿ [Accessed 16 April 2021].

[28] ASUS Global. 2021. [online] Available at: ¡https://www.asus.com/Motherboards-Components/Motherboards/All-series/PRIME-B550-PLUS/¿ [Accessed 16 April 2021].

[29] Hochreiter, S. and Schmidhuber, J., 1997. Long short-term memory. Neural computation, 9(8), pp.1735-1780.