Harnessing Mixture of Experts for Enhanced Abstractive Text Summarization: A Leap Towards Scalable and Efficient NLP Models

Pramod Patil and Akanksha Songire
Department of Computer Engineering, Dr. D. Y. Patil Institute of Technology, Pimpri, Pune, India

Keywords: Abstractive Text Summarization, Mixture of Experts (MoE), Large Language Model, Transformer Architecture, Generative AI, Natural Language Processing (NLP), Deep Learning, Recurrent Neural Network (RNN), Feed Forward Neural Networks (FFNNs).

Abstract: The exploding area of Abstractive Text Summarization (ATS) in Natural Language Processing (NLP) marks a shift from traditional extractive methods, providing more coherent and human-like summaries by creating unique phrases and sentences. While there have been many advancements in ATS in recent years, they encompass unique challenges and opportunities in NLP. The present models encounter concerns like content preservation, factual inconsistency, semantic understanding, etc. This paper outlines an implementation of ATS that adopts the Mixture of Experts (MoE) Model, it improves the efficiency of complex tasks by using multiple small models and activating only necessary ones while processing data. This method enhances the content's quality and the produced output's relevancy. The experiments show that implementing the MoE approach within the framework of ATS improves the content's accuracy and expands the horizons for developing more effective and efficient NLP.

1 INTRODUCTION

We are surrounded by a large amount of information today, knowledge flows from articles, news, social media posts, blogs, and scientific papers. It is a huge amount of information to understand and develop a decision out of it, we need to have insights or process it. However, no human can digest such huge data where Text Summarization becomes the priority. Summarization is mainly done in two ways: Extractive and Abstractive Summarization. In Extractive Summaries as the name suggests it extracts the most relevant and important sentences from the data and generates summaries. In contrast, abstract, summaries are created by rephrasing and rewriting the content from the original text, like the creation of a condensed new version in a fresh way. Sometimes it is desirable to implement a way of presenting facts that differ from those present in the source. For example, the embedding encapsulates what the author intended, more than just quoting and embedding specific parts from the text. With so many other problems in natural language processing (NLP), people have begun to tackle the exciting task of generating an abstract text! Over the years, various approaches have evolved to understand the problem of summarizing text.

Rule-based approaches being the oldest NLP methods, involve applying a set of heuristic rules and constructed features to extract information or capture structures. There are progress statistical approaches where algorithms like Term Frequency - Inverse Term Frequency and latent Semantic Analysis were used for summarization. Even graph-based methods advanced, where sentences were treated as nodes and were connected based on similarity, sentences with higher scores were selected. In recent years the advancements in Machine Learning Approaches increased to overcome issues like capturing long-term dependencies and global content from supervised, unsupervised models to the latest sequence-to-sequence models. With the introduction of sequence-to-sequence models in (Sequence to Sequence Learning with Neural Networks - Sequence to Sequence models by Google), the encoder-decoder model where variable length input is passed to encoder block which turns it into fixed-length tokens and those are sent to the decoder. The power of Recurrent Neural Network (RNN) is leveraged, which made summarized tasks more sophisticated with the masking models to produce smoother and

Patil, P. and Songire, A.
Harnessing Mixture of Experts for Enhanced Abstractive Text Summarization: A Leap Towards Scalable and Efficient NLP Models.
DOI: 10.5220/0013620600004664

In *Proceedings of the 3rd International Conference on Futuristic Technology (INCOFT 2025) - Volume 3*, pages 420-426
ISBN: 978-989-758-763-4

better summaries. The limitations of the seq-to-seq model like processing one word by word and fading gradient were overcome in transformers with an attention mechanism that processes complete sentences in parallel and has an additional attention layer to capture dependencies.

A new Approach unexplored in this domain is using a Mixture of Experts (MoEs), the term was first introduced in 1991(Adaptive Mixture of Local Experts) to have a supervised technique for systems having multiple networks each handling different input space. Between 2010 and 2015 different areas of research contributed to this field, commonly MoEs were thought of as complete systems having Expert layers and routers lately they coined MoEs as components of deep networks making them larger and more efficient. Secondly, the investigation of conditional computation where the dynamic activation and deactivation of parts of the network were managed. MoE, a type of dynamic neural network architecture, incorporates a set of 'experts' or sub-models that perform a specific task depending on the input data requirements. The technique involves a gating mechanism that allows the model to allocate processing power to certain tasks relieving other tasks by directing relevant input parts to the most appropriate experts.

2 RELATED WORKS

A complete study is conducted to learn about existing abstract text summarization systems, identify research gaps, and determine the need for developing an effective and efficient ATS model with high accuracy.

The paper (Zixiang Chen, Yihe Deng, 2022) particularly details the MoE framework, the sparsely connected model that has achieved success and expanded a new variation in neural networks. We certainly delve into this paper to explain why the mixture model does not collapse into one prominent model, and how the MoE layer improves the performance of learning in neural nets. The main conclusion of our empirical results is that the effectiveness of MoE depends mainly on the structure of the underlying problem and the nonlinear nature of the expert. Two scenarios are compared in this work (1) a single expert (i.e. base model) versus a mixture of experts for particular tasks. The authors concluded after conducting tests on toy datasets that the single expert model reached its highest precision at 87.5%, in comparison with which the Mixture of expert models outperformed it and showed increased efficiency. The work also found that the router can learn centric features and divide complex tasks into sub-tasks which can be solved easily by the experts.

Lately, in the paper (Weilin Cai, Juyong Jiang, 2024), there is a detailed survey on a range of advancements and architecture of Mixture of Experts (MoE) models from 2018 to 2024. The two types of experts namely Sparse MoE and Dense MoE are elucidated, and the working and formulation of the gating mechanism in these are demonstrated. Working of routers, the distribution of input towards various experts available, and training of routers to perform the division and allocate the sub-tasks to experts is discussed with various methods like auxiliary losses and load balancing, etc. This survey closes the gap and has been a vital tool for inspecting the complexities of MoE by the researchers. After a brief review of the structure of the MoE layer, the presentation of a new MoE taxonomy is done. The pre-trained models and several variants of core design available to date of research and comparison of those are reviewed, both by algorithmic and systemic elements.

In traditional transformers, FFNNs are used as an internal layer to capture intrinsic patterns of the data, it expands twice the input tokens and then converses to the number of the same tokens again. In the paper (Xu Owen He, 2024) they have tried to overcome the disadvantages of initial architecture that grow linearly with the increase in the width of the hidden layers. The method proposed in the research is called PEER or Parameter Efficient Expert Retrieval, a technique that can be retrieved from large pools. The architecture decouples model size from computing cost by using a sparse experts architecture to effectively exploit more than a million experts. Regarding the performance-compute trade-off, experiments in language modeling tasks suggest that PEER layers are better than these coarse-grained MoEs and dense feedforward layers.

The paper (Gospel Ozioma Nnadi, Favio Bertini, 2024) serves as a base for the work done on abstractive text summarization and the advancements done recently, specifically using neural networks. The work is divided into 5 sections where the authors have discussed the seq-to-seq models, mechanisms, training techniques, and how to optimize the existing models. Detailed description of the encoder-decoder models along with the datasets commonly used for summarization tasks and evaluation metrics are explained. It helps in understanding the artificial neural nets and recurrent neural nets-based models. Mechanisms like attention, copying, distraction, and coverage are used in architecture for summaries

generation using neural nets. Survey (Hassan Shakil, Ahmad Farooq, Jugal Kalita, 2024) details the state-of-the-art architecture and the advancements from traditionally used architectures to the recent forms of transform models. The evaluation and methods used in recent architecture for summarization tasks are discussed in detail. The future improvements possible and the path for researchers to delve deep into abstractive methods for summary creation are discussed in detail.

The paper (Mike Lewis, Yinhan Liu, 2019) introduces BART, the denoising autoencoder for pretraining sequence-to-sequence models. For BART, to learn the model and recover the original text, it is first pre-trained on noisy text using a noise function. This generally follows the conventional Transformer-based design in generalizing BERT (bidirectional encoder) and GPT (left-to-right decoder). On a range of tasks, the model produces state-of-the-art results with gains of up to 3.5 ROUGE, including summarization. It is effective at tasks relevant to text creation, translation, and comprehension. Implementing architecture can be difficult and involves significantly more computer knowledge.

Implemented for direct copying words from source text and the generation of new words in a single pass, this is a hybrid pointer-generator network, which utilizes both the abstractive and the extractive methods. Another aspect of utilizing this technique is that it lets the model rely on a coverage approach to not copy information in circles. When used with the CNN/Daily Mail summarizing task, at least two ROUGE points outperformed the cut at the edge of any earlier attempt. Improves summary by abstractive extracting information together with purely extracting. Quoting the source text reinstates the original facts to be repeated word for word. Over-repetition in the summaries is avoided due to the coverage technique used. The above model raises two problems:

a. The architecture might be relatively easy to optimize and implement.

b. Huge processing power to train.

Based on the BART objective, in the paper (Yinhan Liu, Jiatao Gu, 2020), mBART is proposed as a whitening autoencoder linguistic sequence-to-sequence that pre-training on huge monolingual data in multiple languages. The technique of the mBART forms one of the first preliminary training strategies of a whole sequence-to-sequence model that denoises whole texts in multiple languages. It can be directly fine-tuned for both machine translation tasks supervised at the sentence and document levels as well as unsupervised and shows significant performance gains across most translation tasks.

3 PROBLEM STATEMENT

With the ever-increasing information around the world, the need for summarized content has become a necessity. We need to be time-efficient and precise in the content that we need versus the data that we consume. MoE framework is analyzed for Abstractive Text Summarization to overcome the word-to-word processing and long-distance memory issues of the transformer's architecture, the widely used technique for summarization. The model targets quality and coordination of content production by dynamically directing information input to the network experts, thereby solving the problem of respecting semantic integrity and intelligence in the management. The proposed approach helps to enhance the efficiency and scalability of summarized content with fewer computational requirements.

4 OBJECTIVES

The main aim of the current research is stated below:

1 Build Understanding of Abstractive Text Summary.

2 Study the Mixed Expert (MoE) Approaches.

3 Design Strategies towards text summarization using Awakening of Experts.

4 To assess scalability and efficiency.

5 To outline the areas for enhancement in future studies in the respective sector.

5 METHODOLOGY

5.1 Normal LLM

A Normal LLM forwards all input and parameters (weights and biases) to the chosen base model, such as T5, BART, and the transformers, which are generally FFNNs. They expand internally with 2x the number of tokens. For example, if 512 tokens are passed as input, the FFNN's internal layer expands to 1024 to find the intricate relationships between the tokens and then converses to 512. The underlying model is fine-tuned or pre-trained to fit the particular tasks.

5.2 Mixture of Experts

One such type of neural network architecture designed to increase the effectiveness and efficacy of machine learning activities is called the Mixture of Experts or MoE. This is achieved by breaking down the problem into smaller jobs that experts and highly specialized sub-models then manage. Each one is trained to be an expert in a particular area of the overall task. It improves the overall performance and efficiency of the model by dynamically choosing the most relevant experts for each input through a gating mechanism.

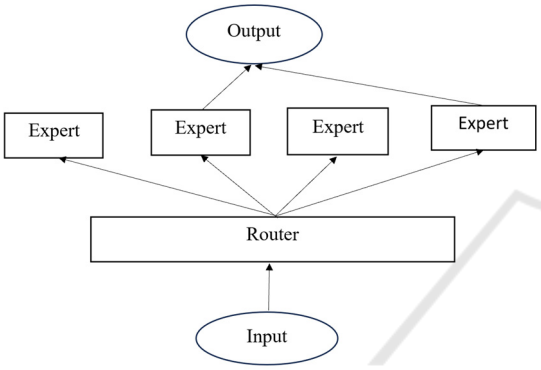

Figure 1: Mixture of Expert

5.2.1 Dense Mix of Experts

Dense Mixture of Experts uses all the experts to process all the inputs, since all the parameters (weights and biases) are passed to FFNNs, they use the gating mechanisms for distribution purposes. It means that every expert thinks over the whole input before their output is combined with that of another expert. This ensures all specialists contribute to the final result produced, although its high computational cost can be a bit unfriendly.

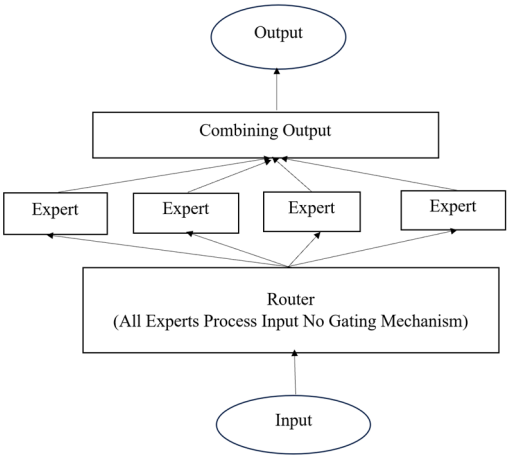

Figure 2: Dense Mix of Experts

5.2.2 Sparse Mix of Experts

The Sparse MoE framework uses the concept of conditional computing, unlike dense models which use all parameters for all inputs, the sparse models activate only some parts of the parameters. This approach of Sparse MoEs helps to scale the size of the model allowing to integrate of thousands of experts.

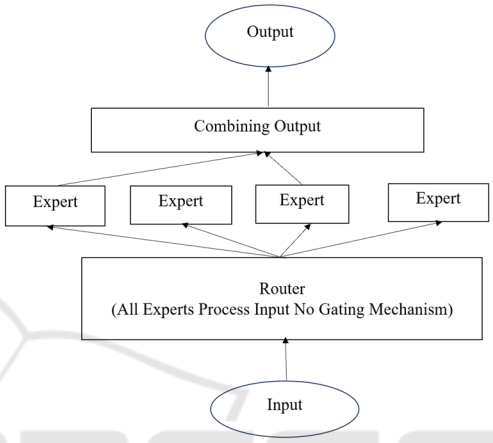

Figure 3: Sparse Mix of Experts

6 PROPOSED MODEL

The mixture of Experts uses different sub-models or experts to improve the quality of LLMs. MoE is defined by two main components namely Experts and Router or gating mechanism. In LLMs there are FFNNs used in each layer, MoE uses a set of experts in each layer which are FFNNs themselves. The router decides which token should be sent to which expert. Experts learn more fine-grained information rather than the whole domain. Since most LLMs have many decoders, the input will go through multiple experts before the actual text generation (each decoder has one expert). The router is trained on, which expert to choose based on a specific input. The output is the probability which helps them to select the best experts. The output of the selected expert is multiplied by the probability of the gate, the expert along with the router makes up the MoE. The gating mechanism is the most important part which decides during inference as well as training phases. The basic form is input multiplied by router weight(W):

$$H(x) = x * W \quad (1)$$

This sets the weights of all but the top 2 to -infinity.

While getting the softmax on the weights -infinity results in probability 0.

6.1 Architecture Diagram

1 Input Layer: Transforms the input text into embeddings to enable further processing by the model.

2 PEGASUS Encoder: The PEGASUS encoder is essentially a multi-layered version of the Transformer that sequentially processes the embedded text. This is where the contextual meaning of the input text is realized.

3 Gating Mechanism: The gating mechanism makes the selection of the most relevant set of experts from a pool of specialized experts on the encoded input.

4 Experts: Each expert analyzes the input they have been provided with, focusing on aspects of the summarizing task a little different from others/contextual understanding, coherence, and factuality.

5 Aggregation of Outputs: For an overall coherent representation of the summary text to be achieved, the active experts' outputs are aggregated.

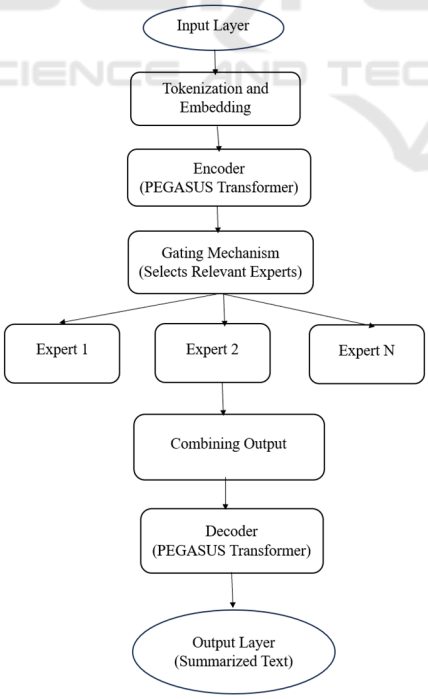

Figure 4: Proposed Model

6 Decoder (PEGASUS): For the final abstractive summary to be acquired, the aggregated representation is fed into the PEGASUS decoder.

7 Output Layer: The output of the decoder is the final condensed text.

6.2 PEGASUS Base Model

1 Summarization-Specific Design: PEGASUS is specifically designed for summarization tasks. It intrinsically lends itself to the task much better than general-purpose language models, as it uses a particular type of pretraining target designed specifically to generate summaries.

2 Gap Sentence Generation (GSG) Pre-Training Method: PEGASUS employs a novel pre-training method called Gap-Sentence Generation, masking out entire phrases and training the model to generate the masked content. This further improves model understanding and summarization abilities as it more closely resembles the task of summarization.

3 State-of-the-Art Performance - Benchmark Results: PEGASUS outperforms other models at each step with significant improvements over a range of summarization benchmarks, including CNN/Daily Mail and XSum.

4 Efficient Use of Training Data - Data Efficiency: It is more efficient than the other models with less amount of data due to the GSG pre-training objective which allows it to make the most out of its training data.

5 Domain Flexibility - Versatility: PEGASUS proved that it can be used over a variety of domains. That makes it eligible for the summarization of varied content, from research publications to news articles. All of this is very crucial to the research application.

6 Compatibility with MoE - Improvement with MoE: By integrating the Mixtures of Experts, PEGASUS's effectiveness in understanding and generating summaries can further be optimized. The MoE's specialization and efficiency improvements will allow PEGASUS to handle more extensive and complex summarization jobs efficiently.

7 Transformer Backbone - Firm Base: PEGASUS, itself founded on the Transformer architecture, heavily relies on the scalability and robust attention mechanisms natively intrinsic within Transformers. Both of those aspects are critical to quality summaries.

MoE techniques significantly enhance the performance and efficiency of the abstractive text summarization process. Following are the benefits of applying various abstractive text summarization experts. In the MoE model, specialists possess

specific knowledge of many features of summarization such as factual accuracy, sentence structure, and contextual meaning. A Gate dynamically decides which experts should be called to activate just the experts that are needed at that particular time for the particular input at hand. The model can save on computation costs with good performance by making use of very few experts. Professional experts working collectively will deliver better accuracy and coherence with the summary produced.

7 RESULTS AND DISCUSSION

Figure 5 and Figure 6 display the performance metrics where the Abstract Summarization is deployed using Normal LLM versus the summarization performed by integrating the base model with the Mix of Experts. A mix of Experts outperforms the tasks in fewer computation costs since there are sub-experts that are selected based on the specialist knowledge they possess. Integrating Mix of Expert in PEGASUS which is the best base model for text summarization tasks also outperforms the work done by normal PEGASUS.

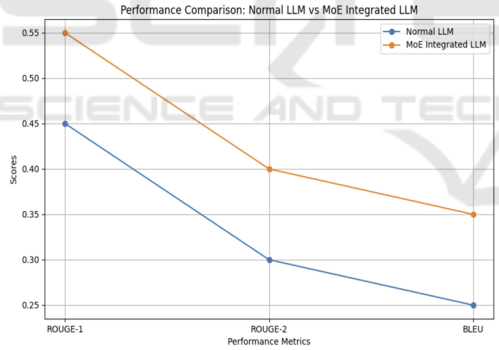

Figure 5: Normal LLM versus MOE

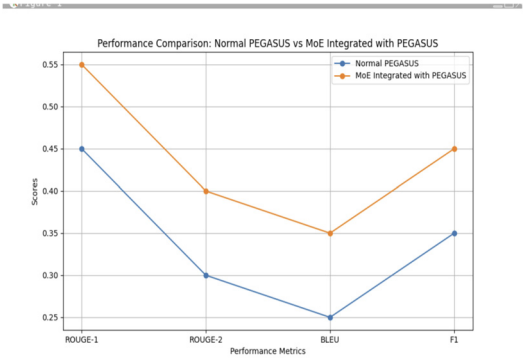

Figure 6: PEGASUS versus PEGASUS with MOE

8 CONCLUSION

The MoE approach, which combines the strength of special-purpose models of experts, appears to be a promising way to extend abstractive text summarization. Summaries generated by the MoE models become not only more accurate and coherent but also more contextually relevant because they consider outputs from multiple experts. This approach can actually help to avoid several more serious drawbacks of the traditional techniques of summarization, such as exposure bias and difficulties arising from large search spaces. But then, the MoE model, as it is called, comes along with its own set of challenges, such as high complexity and increased requirements on resources and also the danger of overfitting. Nevertheless, the Mixture of Experts technique is an invaluable tool in the research and practice of natural language processing, since it brings specific advantages in performance and flexibility in natural language processing. The article concludes by highlighting MoE's dual role in abstractive text summarization; while offering multiple performance and adaptability benefits, it once again illustrates real-world issues that need to be addressed quite effectively. This paper's core insight is its fair play in the strengths and weaknesses of MoE, thus imparting an all-round understanding of the practicality of this model in actual practice.

REFERENCES

Zixiang Chen, Yihe Deng, Yue Wu, Quanquan Gu, and Yuanzhi Li,2022, Published in ArXiv Machine Learning.

Weilin Cai, Juyong Jiang, Fan Wang, Jing Tang, Sunghun Kim, Jiayi Huang, 2024, The publishing journal ArXiv Vol. 14.

Xu Owen He, 2024, Mixture of Million Experts.

Gospel Ozioma Nnadi, Favio Bertini, 2024, Issued in ArXiv Artificial Intelligence.

Hassan Shakil, Ahmad Farooq, Jugal Kalita, 2024, Produced in Computation and Language ArXiv.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, Luke Zettlemoyer,2019, Printed for Transformer's BART model.

Abigail See, Peter J. Liu, Christopher D. Manning, 2017, Brought up on Pointer-Generator Networks

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, Luke Zettlemoyer, 2020, Revised on mBART model.

Ilya Sutskever, Oriol Vinyals, Quoc V. Le, 2014, Published on seq-to-seq neural nets.

David Eigen, Marc' Aurelio Ranzato, Ilya Sutskever, 2014, Communicated on Deep Mixture of Experts ArXiv.
Kumar, S., Solanki,2023, Issued in Springer.
Shubham Dhapola, Siddhant Goel, Daksh Rawat, Satvik Vats, Vikrant Sharma,2024, Publishing company is IEEE 3rd World Conference on Applied Intelligence and Computing (AIC).