Sequence-to-Sequence Networks for Multi-Model Text Summarization

Jefrina Jahangir
University of New Haven
New Haven, CT 06511
jjaha1@unh.newhaven.edu

Radia Zaman Sarin
University of New Haven
New Haven, CT 06511
rsari1@unh.newhaven.edu

Abstract

This project focuses on developing a system for summarizing medical literature using transformer-based models: **PEGASUS**, **BART**, and **BERT**. The goal is to generate concise summaries for medical research articles, specifically from the **medical_cord19** and **medical_meadow_cord19** datasets. Each model was fine-tuned with domain-specific data and evaluated using **ROUGE** metrics. The results show that **BART** achieves the highest ROUGE scores, followed by **PEGASUS**, with **BERT** performing the least effectively. The fine-tuned models are evaluated for their efficiency in summarizing medical content accurately and concisely. The report also includes detailed methodology, training configurations, and results analysis.

1. Introduction

In the field of medical research, the sheer volume of articles and clinical data makes it difficult for professionals to stay updated with current developments. Manual summarization of these articles is not feasible given the time constraints and the repetitive nature of the task. Automatic text summarization using transformer-based models offers a viable solution.

Transformer models like **PEGASUS**, **BART**, and **BERT** have shown significant advancements in natural language processing (NLP), particularly for summarization tasks. These models can be fine-tuned on domain-specific data to generate accurate and concise summaries of medical literature. This project aims to fine-tune these models on two medical datasets and evaluate their performance using **ROUGE** metrics to determine which model is best suited for summarizing medical texts.

2. Objective

2.1 Accurate Medical Text Summarization

The primary objective of this project is to fine-tune transformer-based models to generate accurate and concise summaries of medical literature. This helps in extracting key information from lengthy research articles efficiently.

2.2 Model Performance Comparison

We aim to compare the performance of **PEGASUS**, **BART**, and **BERT** models by evaluating them on two medical datasets using ROUGE metrics.

2.3 Identifying the Best Model

The goal is to identify which model performs best for medical text summarization tasks, providing insights into their suitability for real-world applications in the medical domain.

3. Related Work

3.1 Transformer-Based Models

Transformer models have revolutionized NLP tasks, including summarization. Their ability to process context-rich information makes them ideal candidates for medical text summarization.

3.1.1 PEGASUS

PEGASUS (Pre-training with Extracted Gap-sentences for Abstractive Summarization) is specifically designed for summarization tasks. It uses a gap-sentence generation (GSG) pretraining objective to predict masked sentences.

3.1.2 BART

BART (Bidirectional and Auto-Regressive Transformers) combines the bidirectional encoding of BERT with the autoregressive decoding of GPT. This makes it effective for both text comprehension and generation tasks.

3.1.3 BERT

BERT (Bidirectional Encoder Representations from Transformers) is primarily designed for understanding tasks but can be adapted for summarization with additional fine-tuning.

3.2 ROUGE Metrics

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) is a set of metrics commonly used to evaluate text summarization by measuring the overlap between generated and reference summaries. The key metrics used are:

- **ROUGE-1**: Measures unigram overlap.

- **ROUGE-2**: Measures bigram overlap.
- **ROUGE-L**: Measures the longest common subsequence.
- **ROUGE-Lsum**: Like ROUGE-L but for entire summaries.

4. Datasets

For this project, we used two medical datasets sourced from **Hugging Face**. These datasets were chosen for their relevance to medical research and summarization tasks. Each dataset contains research articles paired with corresponding summaries.

4.1 medical_cord19

- **Description**:
  The **medical_cord19** dataset consists of research articles specifically focused on **COVID-19**. This dataset provides full-text articles and their corresponding summaries, making it suitable for domain-specific summarization tasks.
- **Source**:
  The dataset was sourced from Hugging Face and can be accessed via the following link: medical_cord19 on Hugging Face
- **Size**:
  - **Total Samples**: 500
  - **Training Set**: 400 samples (80%)
  - **Testing Set**: 100 samples (20%)
- **Features**:
  - **input**: The full text of COVID-19-related research articles.
  - **output:** Summaries capturing the key information from each article.

4.2 medical_meadow_cord19

- **Description**:
  The **medical_meadow_cord19** dataset contains research articles covering a broader range of general medical topics. This dataset is valuable for evaluating how well the models generalize to medical literature beyond COVID-19.
- **Source**:
  The dataset was obtained from Hugging Face and is available at the following link: medical_meadow_cord19 on Hugging Face
- **Size**:
  - **Total Samples**: 500
  - **Training Set**: 400 samples (80%)
  - **Testing Set**: 100 samples (20%)
- **Features**:
  - **input:** Full-text medical research articles from various domains.
  - **output**: Corresponding summaries highlighting essential information and findings.

5. Methodology

This section outlines the detailed methodology used to fine-tune the transformer-based models which are **PEGASUS**, **BART**, and **BERT** for medical text summarization. The process covers techniques and libraries employed, preprocessing steps, model descriptions, fine-tuning configurations, and training outcomes.

5.1 Techniques and Libraries

To facilitate the fine-tuning process, we employed several techniques and libraries that are essential for handling data, training models, and evaluating performance.

Techniques

1. **Gradient Accumulation**:
    - Due to GPU memory constraints, gradient accumulation was used to simulate larger batch sizes. Gradients accumulated over multiple steps before updating the model weights, allowing efficient training with small batch sizes.
2. **Dynamic Padding**:
    - Applied dynamic padding to ensure that sequences within each batch were padded to the length of the longest sequence. This technique optimizes memory usage and training efficiency by avoiding unnecessary padding.
3. **Warmup Steps**:
    - Employed warmup steps to gradually increase the learning rate at the beginning of training. This helps stabilize the training process and prevents large gradient updates that could destabilize learning.
4. **Early Stopping**:
    - Configured models to load the best checkpoint based on validation loss to avoid overfitting and ensure the best model is used for evaluation.

Libraries

1. **Transformers** (Hugging Face):
    - Provides pre-trained models, tokenizers, and the Trainer API for fine-tuning transformer models.
2. **Datasets** (Hugging Face):
    - Facilitates loading, preprocessing, and splitting datasets into training and testing sets.
3. **PyTorch**:
    - Used for model training and optimization, providing flexibility and efficiency in handling neural networks.
4. **Accelerate**:
    - Simplifies multi-device and mixed-precision training, ensuring efficient utilization of GPU resources.
5. **DataCollatorForSeq2Seq**:

- Handles dynamic padding and collation of data batches for sequence-to-sequence tasks.

5.2 Preprocessing Steps

Preprocessing is a crucial step to prepare the datasets for fine-tuning. The following steps were applied to both **medical_cord19** and **medical_meadow_cord19** datasets:

1. **Text Cleaning**:
   - **Standardization**: Removed special characters, extra white spaces, and standardized formatting to ensure consistency.
   - **Noise Removal**: Eliminated metadata, references, and irrelevant content to focus on the core text.
2. **Tokenization**:
   - Tokenization converts text into tokens that the models can process. Each model uses a specific tokenizer:
     - **PEGASUS Tokenizer**: Optimized for summarization tasks, splitting text into subwords relevant for the encoder-decoder architecture.
     - **BART Tokenizer**: Preserves bidirectional context for effective comprehension and generation.
     - **BERT Tokenizer**: Uses **WordPiece** tokenization to handle a large vocabulary efficiently.
3. **Train-Test Split**:
   - The datasets were split into **80% for training** and **20% for testing** to evaluate model performance on unseen data.
4. **Dynamic Padding**:
   - Applied dynamic padding to ensure each batch was padded to the length of the longest sequence, improving memory efficiency during training.

5.2 Model Descriptions

5.2.1 PEGASUS

PEGASUS (Pre-training with Extracted Gap-sentences for Abstractive Summarization) is specifically designed for text summarization tasks. It follows an encoder-decoder architecture, where:

- The encoder processes the input text to generate contextual representations.
- The decoder generates the summary based on these representations.

Key Feature:
PEGASUS uses a unique Gap Sentence Generation (GSG) objective during pretraining. In this method, important sentences in the input are masked (treated as "gaps"), and the model learns to generate these masked sentences. This makes PEGASUS particularly effective for summarizing long documents where key sentences encapsulate the core information.

5.2.2 BART

BART (Bidirectional and Auto-Regressive Transformers) combines the strengths of **BERT** and **GPT**:

- **Bidirectional Encoder** (like BERT): Reads the input text in both directions (left-to-right and right-to-left) to capture comprehensive contextual information.
- **Autoregressive Decoder** (like GPT): Generates the output (summary) one token at a time.

Key Feature:
BART is pretrained using a denoising autoencoder objective, where the model learns to reconstruct corrupted text. This makes it highly versatile for summarization tasks, as it can understand the input context deeply and generate coherent summaries.

5.2.3 BERT

BERT (Bidirectional Encoder Representations from Transformers) is an encoder-only model designed primarily for understanding tasks such as classification and question answering. For summarization, BERT was adapted as a sequence-to-sequence model by pairing it with a decoder during fine-tuning.

Key Feature:
BERT uses Masked Language Modeling (MLM) as its pretraining objective, where random tokens are masked, and the model learns to predict them based on the surrounding context. While BERT is not inherently designed for text generation, its strong contextual understanding can still be leveraged for extractive summarization tasks.

5.3 Fine-Tuning Process

Fine-tuning involves updating the weights of the pre-trained models using domain-specific data to adapt them to the summarization task. The process was tailored for each model, with hyperparameters chosen based on the computational constraints and model requirements.

5.3.1 Fine-Tuning PEGASUS

1. **Training Configuration**:
    - **Epochs**: Since PEGASUS is computationally intensive, we fine-tuned it for **1 epoch**.
    - **Batch Size**: Due to GPU memory limitations, the batch size was set to **1**. To simulate a larger batch size, **gradient accumulation** was used, combining gradients over **16 steps** before updating the model weights.
    - **Warmup Steps**: Set up to **500** to gradually increase the learning rate and stabilize training.
    - **Evaluation**: The model was evaluated every **500 steps** to monitor performance.
2. **Training Outcome**:
    - **Dataset 1 (medical_cord19)**:

        - Training Loss: **4.3714**
        - Runtime: **279.996 seconds**
    - **Dataset 2 (medical_meadow_cord19)**:
        - Training Loss: **4.3316**
        - Runtime: **197.4023 seconds**

5.3.2 Fine-Tuning BART

1. **Training Configuration**:
    - **Epochs**: BART was fine-tuned for **3 epochs** to allow the model to learn from the data over multiple iterations.
    - **Batch Size**: Set to **2** due to GPU constraints, with **gradient accumulation over 8 steps** to achieve an effective batch size of 16.
    - **Learning Rate**: A learning rate of **5e-5** was used to balance training speed and model stability.
    - **Evaluation**: Conducted at the end of each epoch to track improvements.
2. **Training Outcome**:
    - **Dataset 1 (medical_cord19)**:
        - Training Loss: **2.2061**
        - Runtime: **475.934 seconds**
    - **Dataset 2 (medical_meadow_cord19)**:
        - Training Loss: **2.4897**
        - Runtime: **585.3142 seconds**

5.3.3 Fine-Tuning BERT

1. **Training Configuration**:
    - **Epochs**: Fine-tuned for **3 epochs** to give BERT sufficient exposure to the data.
    - **Batch Size**: Set to **2**, with **gradient accumulation over 8 steps** to simulate a larger batch size.
    - **Learning Rate**: A learning rate of **5e-5** to ensure stable training.
    - **Evaluation**: Performed after each epoch to monitor training progress.
2. **Training Outcome**:
    - **Dataset 1 (medical_cord19)**:
        - Training Loss: **3.6287**
        - Runtime: **254.6717 seconds**
    - **Dataset 2 (medical_meadow_cord19)**:
        - Training Loss: **3.6091**
        - Runtime: **276.248 seconds**

5.6 Evaluation Metrics

The evaluation of the fine-tuned transformer models (**PEGASUS**, **BART**, and **BERT**) for medical text summarization was conducted using the **ROUGE (Recall-Oriented Understudy for Gisting Evaluation)** metric suite. These metrics were chosen because they are the standard for assessing

the quality of text summaries by measuring the overlap between the generated and reference summaries. The evaluation process involved the following metrics:

1. **ROUGE-1**:
    - Measures the overlap of **unigrams** (single words) between the generated summaries and the reference summaries.
    - Reflects how well the model captures key words present in the reference summaries.
2. **ROUGE-2**:
    - Measures the overlap of **bigrams** (two consecutive words).
    - Assesses the model's ability to produce coherent and contextually accurate phrases.
3. **ROUGE-L**:
    - Measures the **Longest Common Subsequence (LCS)** between the generated and reference summaries.
    - Evaluates the structural similarity and fluency of the generated summaries.
4. **ROUGE-Lsum**:
    - Similar to ROUGE-L, but applied to the entire multi-sentence summaries.
    - Provides insights into the overall coherence and completeness of the generated summaries.

5.7 Evaluation Process

1. **Post-Fine-Tuning Evaluation**:
    - The models were evaluated **after fine-tuning** on both datasets:
        - **Dataset 1**: medical_cord19
        - **Dataset 2**: medical_meadow_cord19
    - The evaluation was performed on the **test sets** of these datasets.
2. **Metrics Computation**:
    - The ROUGE metrics were computed using the Hugging Face evaluate library, which calculates the **Precision**, **Recall**, and **F1 scores** for each metric.
    - The models' summaries were compared against the reference summaries, and the scores were reported for each dataset.
3. **Comparison**:
    - The ROUGE scores were used to compare the performance of the three models (**PEGASUS**, **BART**, and **BERT**).
    - The evaluation allowed for identifying the improvements made by fine-tuning and determining which model generated the most accurate and coherent summaries.

Overall Implications of ROUGE Scores

- **High ROUGE-1 & ROUGE-2**: The model is good at picking out key terms and forming coherent phrases that match the reference summary's lexical choices and short contextual cues.
- **High ROUGE-L & ROUGE-Lsum**: The model not only finds the right words and phrases but also preserves a narrative structure and logical progression like that of the reference. This indicates stronger global coherence and summary organization.

- **Comparing Metrics**: By examining all these metrics together, we gain a holistic view of summary quality, from basic word coverage (ROUGE-1) to phrase-level coherence (ROUGE-2) and global structural fidelity (ROUGE-L and ROUGE-Lsum).

This evaluation approach provided a comprehensive analysis of each model's summarization capabilities, capturing both lexical accuracy and structural coherence.

6. Results and Analysis

This section presents the final evaluation results of three transformer-based models (**PEGASUS**, **BART**, and **BERT**) before and after fine-tuning on two datasets: **medical_cord19** and **medical_meadow_cord19**. The ROUGE metrics (ROUGE-1, ROUGE-2, ROUGE-L, and ROUGE-Lsum) were used to quantify improvements in summary quality. Additionally, we include representative summaries generated by the fine-tuned models, illustrating qualitative differences in the content and clarity of the outputs.

6.1 Comparative ROUGE Scores

The table below displays the ROUGE scores for each model on both datasets, comparing the base model (pre-fine-tuning) against the fine-tuned model. Higher ROUGE scores generally indicate better alignment with reference summaries in terms of lexical and structural similarity.

Model	Dataset		ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-Lsum
Pegasus	medical_cord19	Base	0.238985	0.104768	0.185111	0.184566
		Fine-tuned	0.225052	0.090024	0.181915	0.181877
Pegasus	medical_meadow_cord19	Base	0.244406	0.100007	0.180423	0.201486
		Fine-tuned	0.239664	0.10323	0.181758	0.203918
BART	medical_cord19	Base	0.239091	0.091287	0.205139	0.204514
		Fine-tuned	0.307451	0.148789	0.260546	0.259868
BART	medical_meadow_cord19	Base	0.257552	0.102131	0.218215	0.218733
		Fine-tuned	0.292813	0.139123	0.249064	0.250171
BERT	medical_cord19	Base	0.22626	0.084069	0.168998	0.168967
		Fine-tuned	0.22692	0.088106	0.170491	0.169417
BERT	medical_meadow_cord19	Base	0.226998	0.089608	0.16404	0.164497
		Fine-tuned	0.219643	0.085449	0.15842	0.159239

6.2 Analysis of Findings

1. **Model-Level Improvements**:
    - **BART** exhibits the most substantial improvement upon fine-tuning. Its ROUGE-1 and ROUGE-2 scores, in particular, show a pronounced increase, highlighting enhanced lexical accuracy and phrase-level coherence in its generated summaries. Improvements in ROUGE-L and ROUGE-Lsum further indicate better structural alignment with the reference summaries.
    - **PEGASUS** shows mixed results. On the **medical_cord19** dataset, there is a slight decrease in ROUGE-1 and ROUGE-2 after fine-tuning, suggesting sensitivity to the dataset's domain or potential overfitting. On the **medical_meadow_cord19** dataset, we observe marginal improvements, indicating that fine-tuning can help in more diverse topic settings, albeit modestly.
    - **BERT** reveals minimal gains from fine-tuning. As an encoder-only model initially designed for understanding tasks rather than generation, BERT's ability to produce coherent abstract summaries remains limited. While it does show a slight improvement, the differences are not as pronounced as with BART or PEGASUS.
2. **Dataset Influence**:
    - Models generally perform better or show clearer improvements on **medical_cord19**, possibly due to more focused domain content (e.g., COVID-19-related literature).
    - In contrast, the **medical_meadow_cord19** dataset's broader medical scope may pose additional challenges for the models, limiting the magnitude of improvement from fine-tuning.
3. **Which Model Worked Better?**
   Considering all the metrics and improvements, **BART emerges as the best-performing model**. Its encoder-decoder architecture and pretraining objective align well with the abstractive summarization task, enabling it to produce summaries that are both accurate (high ROUGE-1 and ROUGE-2) and structurally coherent (improved ROUGE-L and ROUGE-Lsum) after fine-tuning.

6.3 Examination of Fine-Tuned Model Summaries

In addition to the quantitative scores, we assessed sample outputs from each fine-tuned model to verify correctness and thematic alignment.

- **Fine-Tuned PEGASUS (Dataset1 Example)**:
  Model Summary: Discusses a cost-benefit analysis of vaccination against four preventable diseases in older US adults.
  Assessment: The summary accurately reflects the main objective and methods from the original text, capturing the economic evaluation focus. While concise, it aligns well with the reference context.
- **Fine-Tuned PEGASUS (Dataset2 Example)**:
  Model Summary: Proposes a hybrid Machine Learning and Verbal Decision Analysis approach for diagnosing dementia in HIV-infected individuals.
  Assessment: The summary identifies the hybrid approach and the target population (HIV-infected individuals), and the mention of multicriteria methods aligns with the dialogue. It is thematically correct and coherent.

- **Fine-Tuned BART (Dataset1 Example)**:
  Model Summary: Summarizes the first-year results of Ukraine's NAQA accreditation process, noting the percentage of accredited, conditionally accredited, and refused programs, and referencing the shift to remote accreditation during the pandemic.
  Assessment: This output is highly accurate and closely matches the key data and challenges mentioned in the dialogue. It demonstrates BART's ability to capture details and contextual shifts (e.g., pandemic-induced changes).
- **Fine-Tuned BART (Dataset2 Example)**:
  Model Summary: Describes RNA nanotechnology and its suitability for various applications, including logic gates and medical uses.
  Assessment: BART accurately encapsulates the essence of RNA nanotechnology's properties and potential applications, reflecting both the content and the structure of the original text.
- **Fine-Tuned BERT (Dataset1 Example)**:
  Model Summary: Identifies factors influencing M-banking adoption in the Gulf region and references the UTAUT model.
  Assessment: While correct in capturing the general theme, BERT's summary is less detailed and more surface-level than BART or PEGASUS. Still, it is thematically on point.
- **Fine-Tuned BERT (Dataset2 Example)**:
  Model Summary: Presents an LDA-based approach to analyze Twitter data for conspiracy theories, focusing on US elections and COVID-19.
  Assessment: BERT's summary is correct but less elaborative. It mentions key concepts (iterative filtering, conspiracy theories) and contexts but lacks the richer detail seen with BART.

6.4 Overall Insights

- The results confirm that fine-tuning can enhance summary quality, with BART showing the most significant improvements in lexical and structural alignment with the references.
- PEGASUS benefits from fine-tuning in certain domains, though improvements are less consistent.
- BERT, while slightly improved, remains less adept at producing rich, coherent summaries compared to the encoder-decoder frameworks.
- The correctness of the generated summaries, as judged qualitatively, supports the quantitative findings: BART's summaries are typically more nuanced and contextually aligned, while PEGASUS offers moderate improvements and BERT's gains are modest.

The ROUGE metrics and the examination of sample summaries collectively indicate that BART is the strongest performer among the three models. Its fine-tuned outputs are both quantitatively better (as per ROUGE scores) and qualitatively more accurate and coherent, reinforcing the conclusion that BART's architecture and pretraining objectives are particularly well-suited for abstractive medical text summarization.

7. Conclusion

In this project, we successfully fine-tuned three transformer-based models which are **PEGASUS**, **BART**, and **BERT**, to perform abstractive summarization on two medical datasets, **medical_cord19** and **medical_meadow_cord19**. The evaluation, based on ROUGE metrics, demonstrated that **BART** achieved the most substantial improvements, producing summaries that were not only lexically accurate but also structurally coherent. While **PEGASUS** showed moderate improvements, particularly on diverse medical topics, **BERT** exhibited limited gains due to its encoder-only architecture, which is less suited for generative tasks. The fine-tuned models generated concise and contextually relevant summaries, as validated by both quantitative scores and qualitative analysis of sample outputs. These findings highlight the potential of fine-tuned transformer models to automate the summarization of medical literature, enabling researchers and healthcare professionals to quickly extract key insights from vast amounts of research data. Although the results are promising, future work could address limitations related to dataset size, model sensitivity to domain specificity, and architectural constraints, paving the way for even more effective summarization solutions in real-world applications.

8. Acknowledgement

We would like to express our sincere gratitude to everyone who contributed to the successful completion of this project. We are also grateful to the open-source community and platforms like **Hugging Face** for providing access to datasets and pre-trained models, which were essential for the fine-tuning process. Special thanks to the developers of the Transformers library and related tools, which facilitated the implementation of our models. Additionally, Special thanks to Professor Khaled Sayed our course instructor, who offered valuable insights and feedback that helped shape the success of this project.

9. Code and Resources

The code for this project, along with the code demo presentation link, is publicly available for reference and use. The code repository can be found on GitHub at the following link.

GITHUB LINK: https://github.com/R-Z-S/Final_Project_NLP

CODE DEMO PRESENTATION LINK: https://youtu.be/ZGQrzB77Hsg

10. References

1. Bhargavi, Y. K., Srinivas, P., Krishna, V. V., Rao, P. S., & Upendhar, N. (2023, September). Text summarization and translation on multimodal data. In AIP Conference Proceedings (Vol. 2754, No. 1). AIP Publishing.
2. Poornima, M., Pulipati, V. R., & Sunil Kumar, T. (2022, February). Abstractive multi-document summarization using deep learning approaches. In Proceedings of Second International Conference on Advances in Computer Engineering and Communication Systems: ICACECS 2021 (pp. 57-68). Singapore: Springer Nature Singapore.

3. Dar, Z., Raheel, M., Bokhari, U., Jamil, A., Alazawi, E. M., & Hameed, A. A. (2024, April). Advanced Generative AI Methods for Academic Text Summarization. In 2024 IEEE 3rd International Conference on Computing and Machine Intelligence (ICMI) (pp. 1-7). IEEE.
4. Asmitha, M., Kavitha, C. R., & Radha, D. (2024, June). Summarizing News: Unleashing the Power of BART, GPT-2, T5, and Pegasus Models in Text Summarization. In 2023 4th International Conference on Intelligent Technologies (CONIT) (pp. 1-6). IEEE.
5. Al-Banna, A. A., & Al-Mashhadany, A. K. (2023, July). Automatic Text Summarization Based on Pre-trained Models. In 2023 Al-Sadiq International Conference on Communication and Information Technology (AICCIT) (pp. 80-84). IEEE.
6. Sarthak, Rishiwal, V., Yadav, P., Yadav, M., Gangwar, S., & Shankdhar, A. (2024). Fine tuning the large language pegasus model for dialogue summarization. International Journal of Information Technology, 1-13.
7. Singh, J., Patel, T., & Singh, A. (2023, August 3). Performance Analysis of Large Language Models for Medical Text Summarization.
8. Lewis, M. (2019). Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. arXiv preprint arXiv:1910.13461.
9. Zhang, J., Zhao, Y., Saleh, M., & Liu, P. (2020, November). Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In International conference on machine learning (pp. 11328-11339). PMLR.
10. Beltagy, I., Lo, K., & Cohan, A. (2019). SciBERT: A pretrained language model for scientific text. arXiv preprint arXiv:1903.10676.