OptiMinBERT: A Comparative Study on the Efficacy of Multitask Versus Specialist Neural Networks

Stanford CS224N Default Project

Paras Malhotra
Department of Computer Science
Stanford University
parasume@stanford.edu

Abstract

In this project, we explore the adaptability and performance optimization of a pre-trained BERT encoder model across three distinct NLP tasks: sentence sentiment classification (SST), paraphrase detection, and semantic textual similarity (STS). We initially employed a round-robin multitask classifier that was trained on all 3 tasks. To address the inherent challenges of multitask learning, such as task interference and training inefficiency, we integrated gradient surgery techniques into our approach, following the methodologies suggested by Yu et al. (2020). To improve the accuracy of our multitask model, we added a shared sentence embedding layer based on the methodologies by Reimers and Gurevych (2019) by training the embedding layer on cosine similarity loss. Subsequently, we shifted our focus towards the development of specialized neural networks, tailored to each specific task, to harness the full potential of task-specific optimizations. Our empirical results reveal that these specialized networks outperformed the multitask learning approach, demonstrating marked improvements in precision and effectiveness across all tasks. This finding underscores the significant advantages of employing specialized models for individual NLP tasks over a generalized multitask framework, aligning with recent studies that highlight the efficacy of task-specific fine-tuning in the realm of transfer learning such as by Weiss et al. (2016).

1 Key Information to include

- Mentor: Hamza El Boudali
- External collaborators / mentors / sharing: N/A

2 Introduction

The transformative power of deep learning in the field of Natural Language Processing (NLP) has been significantly amplified by the advent of transformer architectures and large pre-trained models like BERT. These advancements have ushered in a new era of performance benchmarks across a wide array of NLP tasks, from sentiment analysis to paraphrase detection and beyond, showcasing remarkable abilities in understanding and generating human-like text. Despite these successes, the nuanced adaptation of such models to specialized tasks remains a challenging frontier. This challenge is not only technical, involving the fine-tuning of complex models on task-specific datasets, but also conceptual, requiring an understanding of how to best leverage shared knowledge across different tasks without diluting the model's expertise in any single domain.

In this research, we use pre-trained embeddings from a BERT encoder model for three specialized tasks: sentence sentiment classification (SST), paraphrase detection, and semantic textual similarity (STS). We first evaluate the raw effectiveness of BERT embeddings for the task of sentiment analysis

without introducing additional model complexities and simply applying a linear transformation to classify sentences according to their sentiment.

Building upon this foundational work, we then integrate gradient surgery techniques by Yu et al. (2020) to enhance the training efficiency of our model, particularly when addressing the multifaceted challenge of multitask learning. This method, designed to optimize the simultaneous learning across diverse tasks, helps to mitigate the potential for task interference and conflicting gradients by projecting the task's gradient onto the normal plane of the gradient of the other task(s).

To improve our model's performance, we add a shared sentence embedding layer suggested by Reimers and Gurevych (2019) that allows the three tasks to benefit from commonalities in the sentence representations derived from BERT. This shared layer is trained on cosine similarity loss and the cosine similarity is used as the direct output for the semantic textual similarity (STS) task, while also fed as input to the paraphrase detection and sentiment analysis (SST) tasks.

Finally, recognizing the limitations of a purely shared approach, we develop specialized neural networks tailored to each specific task. These specialized networks have the same architecture (and neural network layers) as the multitask classifier for each individual tasks but unlike the multitask classifier, the individual tasks do not share weights amongst them and are trained solely on the individual training datasets and task objectives.

Our empirical results reveal that these specialized networks outperformed the multitask learning approach, demonstrating marked improvements in precision and effectiveness across all tasks. These findings underscore the significant advantages of employing specialized models for individual NLP tasks and highlight the efficacy of task-specific fine tuning in the realm of transfer learning. We finally also explore avenues for future work and discuss several enhancements that could be made to improve our models' performance across the various tasks.

3 Related Work

The exploration of multitask learning and the optimization of pre-trained models like BERT for specific NLP tasks have been subjects of increasing interest within the field of natural language processing (NLP). Ruder (2017) discusses soft and hard parameter sharing approaches for multi-task learning. We found hard parameter sharing more appealing for our specific project because it reduces the risk of overfitting and we observed that our baseline sentiment classifier model suffered from overfitting, where additional training was leading to a better training accuracy but a lower accuracy on our development dataset.

To increase our model training efficiency, we integrated the pioneering gradient surgery technique known as PCGrad (Projecting Conflicting Gradients), developed by Yu et al. (2020). This method emerges from the identification of three core challenges in multi-task optimization: conflicting gradients, high positive curvature, and large differences in gradient magnitudes. PCGrad adeptly mitigates these challenges by altering gradients directly when they are found to be conflicting—defined as pointing in opposite directions, thereby impeding progress due to negative cosine similarity. The technique projects each conflicting gradient onto the normal plane of the other, effectively neutralizing the adverse effects of interference and fostering a more harmonious optimization process.

We then shifted our focus on improving the performance of our multitask model. Ruder (2017) discusses that closely related tasks can benefit from shared layers. Taking inspiration from Reimers and Gurevych (2019), we decided to introduce a shared sentence embedding layer across all three tasks. We used the suggested Siamese network structure, where we concatenated the sentence embeddings u and v with the element-wise difference $\mathbf{u} - \mathbf{v}$ for inputs to the paraphrase detection task, and used cosine similarity between the sentence embeddings for the semantic textual similarity task.

Finally, when we developed specialized neural networks tailored to each specific task, we studied the work done by Howard and Ruder (2018). They discuss a spectrum of fine tuning strategies ranging from gradual unfreezing of layer weights and triangular learning rates to a simple transfer technique that employs additional feed-forward layers on top of the pre-trained models. Given the close relationship among our tasks — sentence sentiment classification, paraphrase detection, and semantic textual similarity — we hypothesized that the integration of additional layers would likely offer the most effective method for enhancing task-specific performance.

4 Approach

Our approach to enhancing the performance of a pre-trained BERT model for diverse NLP tasks includes developing a baseline model, a round-robin multitask classifier with a shared sentence embedding layer, and integrating gradient surgery techniques to mitigate task interference. We further advanced our model by implementing specialized neural networks for each task, drawing on fine-tuning strategies such as additional feed-forward layers for task-specific optimization. Below, we detail these components and discuss our approach for each of them.

4.1 Baseline Model

In part 1 of the project, we focused on establishing a solid understanding and foundation of the BERT model's internal workings. We successfully implemented the BERT self-attention layer and the BERT transformer layer from scratch, ensuring our implementation's correctness through passing the predefined tests. This foundational work laid the groundwork for our subsequent development of a baseline model for sentiment classification.

Building upon this, we developed a sentiment classifier leveraging the pre-trained BERT model. Specifically, we utilized the pooled output of BERT's CLS token, which serves as a summary representation of the input sequence's entire context. This representation was then fed into a simple linear transformation layer to classify sentences according to their sentiment. By applying this straightforward approach, we aimed to evaluate the raw effectiveness of BERT embeddings for the task of sentiment analysis without introducing additional model complexities.

4.2 Round-Robin Multitask Classifier

To leverage BERT's capabilities across diverse NLP tasks, we extended the skeleton code, which was initially designed to train solely on SST data, to a multitask learning framework. This framework facilitates simultaneous training on three distinct datasets: SST for sentiment classification, Quora Question Pairs for paraphrase detection, and the SemEval dataset for semantic textual similarity.

Given the disparity in dataset sizes, we employed a round-robin training methodology. This approach ensures equitable learning across tasks by cyclically iterating through datasets with fewer examples, thereby balancing the model's exposure to each task. Furthermore, we tailored the loss functions to suit the nature of each task: cross-entropy loss for SST, binary cross-entropy for paraphrase detection, and mean squared error for STS. This customization reflects our understanding of the inherent differences in task objectives and evaluation metrics.

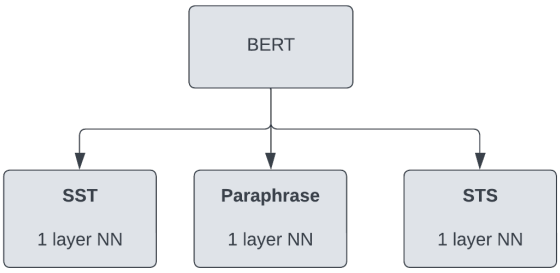

Figure 1: Initial architecture of multitask classifier

4.3 Shared Sentence Embedding Layer

A pivotal enhancement in our model is the introduction of a shared sentence embedding layer based on Reimers and Gurevych (2019). Recognizing that BERT's output, particularly the pooled CLS token, might not directly serve as an optimal sentence representation for all tasks, we devised a single-layer neural network to transform this output into a more task-agnostic sentence embedding. This shared embedding underpins the task-specific components of our model:

- For **STS**, the model computes cosine similarity directly from these embeddings, leveraging their numerical properties to assess sentence similarity.
- In **paraphrase detection**, a neural network processes both embeddings and their absolute difference, capturing nuanced relational features.
- **Sentiment classification** integrates the raw BERT pooled output with the derived sentence embeddings, acknowledging the complex nature of sentiment as a task requiring both semantic understanding and domain knowledge.

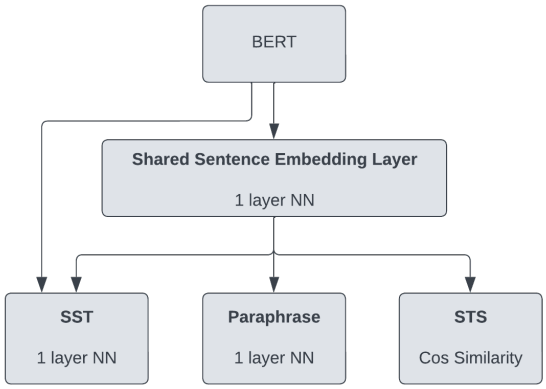

Figure 2: Addition of sentence embedding layer to multitask classifier

We used cosine embedding loss to train the sentence embedding layer. In hindsight, Henderson et al. (2017) suggest that multiple negative rankings loss would have been a much more efficient loss function to train on, where all pairs of sentence inputs are used for training leading to a quadratically larger training corpus. Furthermore, Solatorio (2024) discuss an even more novel and effective approach of GISTEmbed loss, which when compared to multiple negative rankings loss, uses a guide model to guide the in-batch negative sample selection, thereby reducing the reliance of random sampling and improving model accuracies at the expense of some training overhead. We believe that changing our loss function can substantially improve our sentence embeddings and overall model performance.

4.4 Gradient Surgery

Gradient surgery by Yu et al. (2020) is a technique designed to enhance the training process in multitask learning environments. It addresses the issue of conflicting gradients, where simultaneous gradient descent on multiple tasks can lead to suboptimal convergence or even divergence. This is particularly crucial when tasks have different learning objectives that may not align. Through gradient surgery, specifically Projecting Conflicting Gradients (PCGrad), we project each task's gradient onto the normal plane of the other task's gradient when they conflict, thereby resolving the conflicts and allowing for a more cooperative convergence.

When the dot product of a pair of task gradients is negative (i.e., they are conflicting), we use the following projection formula to correct the task gradient:

$$\mathbf{g}_i^{PC} = \mathbf{g}_i^{PC} - \frac{\mathbf{g}_i^{PC} \cdot \mathbf{g}_j}{\|\mathbf{g}_j\|^2} \mathbf{g}_j, \tag{1}$$

where $\mathbf{g}_i^{PC}$ and $\mathbf{g}_j$ are the gradients of tasks i and j respectively.

In our application to the multitask classifier, gradient surgery was used to optimize the shared sentence embedding layer, ensuring that each task's updates contributed positively to the overall learning objective without undermining the performance on other tasks.

5 Experiments

5.1 Data and Evaluation Metrics

This section provides an overview of these datasets, their respective splits and evaluation metrics used.

Stanford Sentiment Treebank (SST) The SST dataset includes 11,855 sentences extracted from movie reviews. It features a rich annotation of 215,154 phrases from parsed trees, each assigned a sentiment label ranging from negative to positive. The dataset is partitioned into training (8,544 examples), development (1,101 examples), and test sets (2,210 examples). We used accuracy as the evaluation metric.

Quora Dataset This dataset contains 400,000 question pairs, with binary labels indicating whether the questions are paraphrases of each other. We were provided subsets for training (141,506 examples), development (20,215 examples), and testing (40,431 examples), with accuracy as the evaluation metric.

SemEval STS Benchmark Dataset The SemEval STS Benchmark dataset includes 8,628 sentence pairs, each scored from 0 (unrelated) to 5 (equivalent meaning). The dataset is split into training (6,041 examples), development (864 examples), and test sets (1,726 examples), with the Pearson correlation coefficient used for evaluation.

Our minBERT model is pre-trained on Wikipedia articles with masked language modeling and next sentence prediction tasks. For downstream task fine-tuning, we employ the aforementioned SST, Quora, and STS datasets, alongside the CFIMDB dataset for baseline comparisons. These datasets necessitate minimal pre-processing such as tokenization, lower-casing, punctuation standardization, and sentence padding for matrix operations.

5.2 Experimental Details

To maintain consistency across our models, we set the learning rate to 1×10^{-3} for pretraining and opted for either 1×10^{-5} or 2×10^{-5} during finetuning. Each model underwent 10 epochs of training with a dropout rate of 0.3 across all hidden layers, and a standard batch size of 64 was used whenever possible (except the baseline model CFIMDB dataset, where a batch size of 8 was used).

The Adam optimizer was utilized without weight decay, featuring correction bias, an epsilon value of 1×10^{-6}, and beta values of 0.9 and 0.999.

5.3 Results

For the baseline model, we achieved the following results:

Phase	Train (Actual)	Dev (Actual)	Dev (Ref)
Pretraining (SST)	0.415	0.396	0.390 (0.007)
Fine-tuning (SST)	0.842	**0.523**	**0.515 (0.004)**
Pretraining (CFIMDB)	0.771	0.771	0.780 (0.002)
Fine-tuning (CFIMDB)	0.999	**0.967**	**0.966 (0.007)**

Table 1: Baseline model results

In Table 1, we report the dev dataset performance of our baseline pre-trained and finetuned BERT models for the baseline sentiment classification model vis-a-vis benchmarks from the project handout. After pretraining and fine tuning, our baseline model closely aligned with the provided reference accuracies. This outcome not only validated our implementation but also established a benchmark for comparing the effectiveness of our enhanced multitask learning model.

Phase	Dev (SST)	Dev (Paraphrase)	Dev (STS)	Dev (Overall)
Round-robin MTL	0.520	0.475	(-0.005)	**0.330**
MTL with sentence embeddings	0.511	0.697	0.517	**0.575**
Specialist neural networks	0.518	0.743	0.616	**0.690**

Table 2: Dev Set Performance Results

Phase	Test (SST)	Test (Paraphrase)	Test (STS)	Test (Overall)
Specialist neural networks	0.527	0.745	0.560	**0.684**

Table 3: Test Set Performance Results

In Table 2, we report the dev set performance for all proposed model approaches/stages of the project. Table 3 mentions the test set performance results. Note that we only evaluated our specialist neural networks on the test set and hence, only reported those results here.

Our quantitative results, as detailed in Tables 1, 2, and 3, provide insightful perspectives on the efficacy of our approaches across different stages of model development and testing. The baseline model performances, both in pretraining and fine-tuning phases, were closely aligned with the reference accuracies provided, confirming the reliability of our implementation and setting a robust benchmark for subsequent comparisons.

The outcomes observed from the round-robin multitask learning (MTL) approach, while promising, did not meet our expectations, particularly in the domain of semantic textual similarity (STS), where the performance marginally regressed. This underperformance highlights the inherent challenge in balancing learning across diverse tasks, suggesting that a uniform application of MTL might dilute the focus required for tasks with more nuanced distinctions.

Conversely, the integration of sentence embeddings into our MTL framework led to a notable improvement in the dev set performance, especially for the paraphrase detection task. This enhancement underscores the value of shared representations in capturing semantic relationships more effectively, thereby validating our hypothesis about the benefits of leveraging sentence embeddings for closely related NLP tasks.

The most significant advancements were realized through the deployment of specialist neural networks. Both on the dev and test sets, these models outperformed their MTL counterparts across all tasks, with particularly impressive gains in paraphrase detection and STS. This success reaffirms the importance of task-specific tuning and model specialization, providing compelling evidence that a focused approach can yield superior results compared to generalized multitask frameworks.

In conclusion, our exploration into multitask learning and the application of specialized models for distinct NLP tasks reveals the intricate balance required between shared knowledge utilization and task-specific optimizations. The marked improvement observed with specialist neural networks indicates that tailored approaches significantly enhance model performance. Based on these insights, we posit that further refinements, such as adopting a Multiple Negatives Ranking Loss or GISTEmbed Loss for our loss function, could potentially yield an additional performance boost, possibly in the realm of 10%.

6 Analysis

In addition to our quantitative evaluation, a *qualitative evaluation* of our models provides deeper insights into their behavior, effectiveness, and areas for improvement. Through examining specific outputs and characteristics, we seek to understand the underlying mechanisms of our system, its strengths, and its limitations.

6.1 Understanding Model Outputs

By scrutinizing the outputs of our models, particularly in cases where their predictions diverged from the expected results, we gained valuable insights into their operational nuances. For instance, the specialist neural networks demonstrated remarkable accuracy in paraphrase detection tasks, often

successfully identifying subtle semantic similarities and differences between sentences. However, in instances of highly nuanced or context-dependent meanings, even these specialized models occasionally faltered, highlighting the challenge of capturing the full complexity of human language.

6.2 Successes and Failures

Our analysis revealed that the models' successes often stemmed from their ability to leverage detailed sentence embeddings and task-specific fine-tuning to capture a wide range of linguistic features. In contrast, failures typically occurred in scenarios involving ambiguous expressions, idiomatic language, or sentences requiring extensive world knowledge for accurate interpretation. Such cases suggest that while our models are adept at handling structured linguistic tasks, they are sometimes limited by the inherent constraints of their training data and the current state of natural language understanding technology.

6.3 Implications for Further Development

This qualitative evaluation underscores the importance of diverse and comprehensive training datasets, the potential benefits of integrating external knowledge sources, and the need for ongoing refinement of model architectures to better capture the subtleties of human language. Furthermore, the insights gained from analyzing model successes and failures guide us toward more nuanced loss functions and optimization strategies, such as the proposed shift to Multiple Negatives Ranking Loss or GISTEmbed Loss, which could further enhance model performance.

In summary, our qualitative analysis not only complements our quantitative findings but also illuminates the path forward for refining our models. It highlights the critical balance between leveraging generalized language understanding capabilities and honing in on the specificities of individual tasks, setting the stage for future advancements in NLP technology.

7 Conclusion

Throughout this project, we embarked on an extensive exploration of the adaptability and performance optimization of a pre-trained BERT model across a variety of NLP tasks, including sentence sentiment classification, paraphrase detection, and semantic textual similarity. Our journey led us through the integration of advanced techniques such as gradient surgery for mitigating task interference, the employment of shared sentence embeddings to foster a deeper understanding across tasks, and the deployment of specialized neural networks tailored to maximize task-specific performance.

Our findings reveal that while multitask learning frameworks offer valuable pathways for leveraging shared linguistic features, the pinnacle of performance is achieved through specialized models that are finely tuned to the unique demands of each task. Notably, the introduction of gradient surgery and task-specific embeddings significantly enhanced our models' ability to navigate the complex landscape of NLP challenges, culminating in a marked improvement in accuracies across all tasks. These successes highlight the critical importance of model customization and the potential of specialized architectures in advancing the field of natural language processing.

However, our work is not without its limitations. The nuanced nature of language and the broad spectrum of linguistic phenomena present challenges that our current models occasionally struggle to fully capture. Ambiguities, idiomatic expressions, and context-dependent meanings remain areas where our systems can see further improvement.

Looking to the future, several promising avenues present themselves for advancing our work. Exploring alternative loss functions, such as Multiple Negatives Ranking Loss or GISTEmbed Loss, offers the potential for even greater performance enhancements. Additionally, the adoption of more advanced pre-trained encoder models and the expansion of our multitask learning framework to include more diverse datasets could further refine our understanding and processing of natural language.

In conclusion, this project not only advances our comprehension of the capabilities and challenges associated with deploying BERT for NLP tasks but also sets the stage for future innovations in model optimization, task-specific tuning, and the exploration of new methodologies in the quest for more sophisticated and nuanced language understanding systems.

References

Matthew Henderson, Rami Al-Rfou, Brian Strope, Yun-Hsuan Sung, László Lukács, Ruiqi Guo, Sanjiv Kumar, Balint Miklos, and Ray Kurzweil. 2017. Efficient natural language response suggestion for smart reply. *arXiv preprint arXiv:1705.00652*.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *CoRR*, abs/1908.10084.

Sebastian Ruder. 2017. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*.

Aivin V Solatorio. 2024. Gistembed: Guided in-sample selection of training negatives for text embedding fine-tuning. *arXiv preprint arXiv:2402.16829*.

Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. 2016. A survey of transfer learning. *Journal of Big data*, 3:1–40.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. 2020. Gradient surgery for multi-task learning. *CoRR*, abs/2001.06782.