Mitigating the Capacity Gap in Knowledge Distillation via Iterative Tutoring

Sara Karam
E.C.E. department
Lebanese American University
Byblos, Lebanon
sara.karam01@lau.edu

Ralph Aouad
R&I department
InMind .ai
Beirut, Lebanon
ralph.aouad@inmind.ai

Joseph Attieh
Dept. of Digital Humanities
University of Helsinki
Helsinki, Finland
joseph.attieh@helsinki.fi

Joe Tekli
E.C.E. department
Lebanese American University
Byblos, Lebanon
joe.tekli@lau.edu.lb

***Abstract*—Large language models (LLMs) have resulted in significant improvements in understanding and generating natural language. However, their deployment in resource-constrained environments is limited by their high computational demands. Hence, Knowledge Distillation (KD) has emerged to address such challenges by enabling the transfer of knowledge from a large, pre-trained model (teacher) to a smaller, more efficient model (student). Yet, some bottlenecks exist in the effectiveness of this technique, such as the "capacity gap" between the teachers' learning abilities and that of the student models, which may negatively impact the distilled model. We address this limitation by introducing a Tutor-Enhanced Iterative Distillation (TEID) to fill the capacity gap, by adding an intermediate-sized tutor model and selective learning strategy to the traditional distillation setup. To achieve further compression, the TEID is repeated iteratively on the tutor and the previously resultant student, with a new smaller student model. Empirical results on the GLUE benchmark show results in mitigating the model capacity gap, while showcasing the need to improve the efficiency and scalability of the distilled models.**

Keywords—Knowledge distillation, Large language models, Capacity gap, Tutor-enhanced model.

1. Introduction

Large language models (LLMs) have revolutionized the field of natural language processing (NLP) by achieving unprecedented performance in various tasks such as text classification, sentiment analysis, and language inference [15, 16]. However, their deployment in real-world applications is significantly hindered by their substantial computational demands and the increasing size of these models as advancements in the field progress [1, 6]. This presents a formidable challenge, especially in resource-constrained environments where computational resources are limited. Knowledge Distillation (KD) has emerged as a promising technique to mitigate this issue by training a smaller, more efficient model (the student) to replicate the performance of a larger, pre-trained model (the teacher) [4, 14].

Despite the effectiveness of KD in reducing the computational load of LLMs, the process is often impeded by the capacity gap between the student and teacher models. This gap refers to the disparity in complexity and learning capacity between the two models, making it challenging for the student to accurately mimic the behavior of the teacher [3, 5]. Previous research has identified this capacity gap as a critical bottleneck, affecting the performance of the distilled models and limiting the efficiency gains from the distillation process [9, 17]. Various strategies have been proposed to address this issue, including architectural adjustments to the student model, modifications to the training procedures, and the use of intermediate representations from the teacher [2, 16].

To address the capacity gap in KD, this research introduces a novel distillation approach featuring a three-tiered hierarchy comprising a teacher, a tutor, and a student model. This Tutor-Enhanced Iterative Distillation (TEID) method incorporates an intermediate tutor model that bridges the gap between the teacher and student, facilitating improved knowledge transfer. The tutor model, being smaller than the teacher but larger than the student, acts as an intermediary, smoothing the transition of knowledge and enhancing the learning capacity of the student model [1, 12]. In addition to the three-tiered hierarchy, the proposed methodology employs a selective learning strategy, where the student model learns from either the teacher or the tutor based on the effectiveness of the knowledge transfer [6]. The approach dynamically adjusts the source of supervision for the student model, optimizing the training process and ensuring that the student receives the most beneficial learning signals [10, 13]. Furthermore, the TEID method introduces a continuous updating and re-distillation process. In this iterative approach, the tutor model is continuously refined and used as a new teacher to further compress a new student model, potentially leading to more efficient and effective model compression over multiple iterations. This continuous refinement aims to progressively enhance the performance of the distilled models, making them more suitable for deployment in resource-constrained environments [13, 14]. Empirical results on the GLUE benchmark show that TEID mitigates the gap of model capacity and improves the efficiency and performance of distilled models. By addressing the critical challenges in KD for LLMs, this work represents a significant advancement in the field, offering a scalable solution for deploying state-of-the-art language models in real-world applications with limited computational resources.

The remainder of this paper is organized as follows. Section 2 provides background and preliminary notions. Section 3 reviews related works, while Section 4 describes our TEID proposal. Section 5 presents the experimental results, before concluding in Section 6 with future works.

2. Related Works

The capacity gap issue in KD has been addressed by various approaches to bridge the learning capacity differences between teacher and student models. Here, we discuss related works revolving around the capacity gap in KD, dynamic distillation, teacher selection, and multi-tier systems, which provide insights relevant to our proposed solution.

2.1. Knowledge Distillation

Knowledge distillation (KD) is a model compression technique that involves transferring knowledge from a large, pre-trained model (the teacher) to a smaller, more efficient model (the student). The student is trained to mimic the teacher's behavior, aiming to retain much of the teacher's performance while reducing computational requirements.

[1] C. Aoun is co-affiliated with the Lab-STICC, CNRS UMR, ENSTA (Ecole Nationale Supérieure de Techniques Avancées), Brest, France

Dynamic knowledge distillation (DKD) frameworks, e.g., [7, 18], improve upon traditional KD by adapting the learning process to the evolving competency of the student model [13], which result in improved performance and training efficiency. The DKD framework introduces three key adjustments: teacher model adoption, data selection, and KD objective adaptation. *Teacher Model Adoption*: Unlike traditional static KD, where the teacher model remains the same throughout the training process, DKD dynamically selects the teacher model based on the student's evolving competency, providing the student with appropriate supervision as it evolves. The framework shows that selecting a smaller teacher initially and transitioning to a larger teacher later improves the student's performance [1]. *Dynamic Data Selection*: DKD frameworks also dynamically select training data based on the uncertainty of the student's predictions, prioritizing challenging instances to enhance efficiency and effectiveness, achieving comparable results with fewer samples [3]. *KD Objective Adaptation*: The supervision signals from different KD objectives (e.g., aligning prediction probabilities and intermediate representations) are dynamically adjusted throughout the training process, to provide the student with the most relevant training signals at each stage of its development [2].

2.2. Capacity Gap in Knowledge Distillation

The capacity gap in KD refers to the discrepancy between the learning capacities of the teacher and student models. This gap can lead to ineffective knowledge transfer and suboptimal performance of the distilled model. Several papers address this issue and propose solutions to mitigate the capacity gap. Residual KD (RKD) is one such method that introduces an assistant model that learns the residual error between the feature maps of the student and teacher models, improving the student's performance without increasing the total computational cost [4]. The RKD method demonstrates superior performance on datasets like CIFAR-100 and ImageNet by effectively narrowing the performance gap between the student and teacher models. The authors in [3] propose a technique called KD via Weighted Ensemble of Teaching Assistants (TAKD), which uses intermediate teaching assistants to progressively transfer knowledge from teacher to student, making learning more manageable and effective, thus addressing the capacity gap. The authors in [9] discuss hint-based training which leverages intermediate feature representations or hints from the teacher to guide the student, improving learning and reducing the impact of the capacity gap. Overall, addressing the capacity gap in KD is crucial for the effective deployment of LLMs in resource-constrained environments.

2.3. Teacher Selection Strategies

Effective teacher selection strategies is critical in KD, as the size and quality of the teacher model can impact the effectiveness of knowledge transfer [1]. *Uncertainty-Based Teacher Adoption*: Techniques that dynamically select teacher models based on the student's prediction uncertainty have been proposed. These methods ensure that the student learns from the most appropriate teacher model at each stage of training, thereby improving the overall performance of the student model. *Teacher Size and Quality*: Studies have shown that a larger teacher does not always result in a better student. The competency of the student and the capacity gap between the teacher and student must be considered when selecting a teacher model. Properly matching the teacher and student models' capacities can lead to better knowledge transfer and improved performance.

2.4. Multi-Tier Knowledge Distillation

Multi-tier Knowledge Distillation (KD) methods address the capacity gap by introducing intermediary models, such as tutors, to provide smoother, stepwise knowledge transfer. For instance, the Tutor-Enhanced Iterative Distillation (TEID) approach, for example, places a tutor between teacher and student to progressively bridge performance disparities [2]. This process follows an iterative distillation strategy, where the tutor is refined and then serves as the teacher for training progressively smaller students, enabling efficient compression over multiple iterations. By allowing students to learn in manageable stages, intermediate tutor models enhance both the effectiveness and efficiency of KD, demonstrating significant performance gains in resource-constrained environments.

In summary, dynamic teacher selection and multi-tier distillation address the capacity gap in KD by adapting the learning process and choosing suitable teacher models to improve effectiveness and efficiency.

3. Tutor-Enhanced Iterative Distillation

This research introduces Tutor-Enhanced Iterative Distillation (TEID) to address the capacity gap in traditional KD for LLMs (cf. Figure 2). TEID employs a teacher–tutor–student setup, where the intermediate-sized tutor bridges the large teacher and smaller student, enabling more gradual and efficient knowledge transfer.

The key components of the TEID framework are as follows (cf. Figure 1): i) *Teacher Model*: A large, pre-trained model that serves as the primary source of knowledge, ii) *Tutor Model*: An intermediate-sized model that learns from the teacher and, in turn, aids the learning, as a stepping stone for knowledge transfer, and iii) *Student Model*: A smaller model that learns from both the teacher and the tutor.

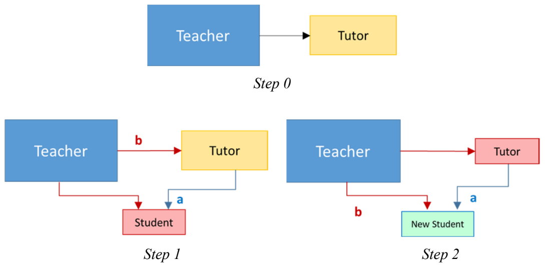
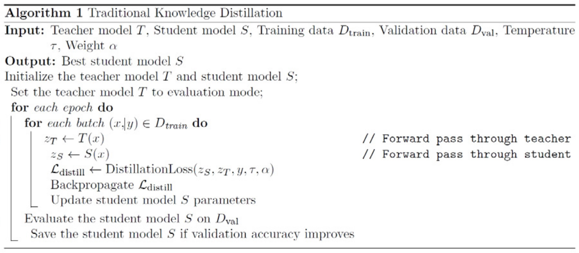


Figure 1. TEID process overview

3.1. Selective Learning Strategy

A cornerstone of the TEID framework is its selective learning strategy, allowing the student to dynamically choose to learn from either the teacher or the tutor, based on which source offers better performance for a given batch of data. It operates as follows: i) For each batch of training data, the student model computes the distillation loss using the outputs from both the teacher and tutor models, ii) The loss values are compared, and the student model updates its parameters based on the model (teacher or tutor) that provides the lower loss, thereby offering better guidance.

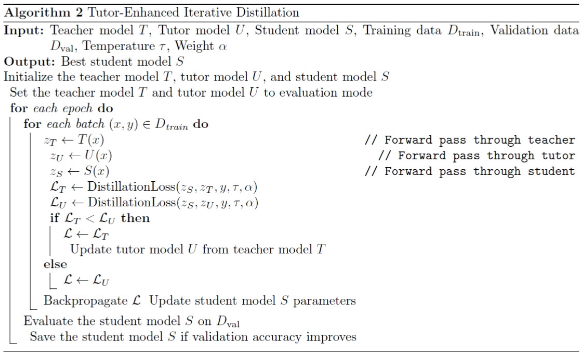


```
Algorithm 1 Traditional Knowledge Distillation
Input: Teacher model $T$, Student model $S$, Training data $D_{\text{train}}$, Validation data $D_{\text{val}}$, Temperature
       $\tau$, Weight $\alpha$
Output: Best student model $S$
Initialize the teacher model $T$ and student model $S$;
 Set the teacher model $T$ to evaluation mode;
 for each epoch do
    for each batch $(x,|y) \in D_{train}$ do
        $z_T \leftarrow T(x)$                          // Forward pass through teacher
        $z_S \leftarrow S(x)$                          // Forward pass through student
        $\mathcal{L}_{\text{distill}} \leftarrow \text{DistillationLoss}(z_S, z_T, y, \tau, \alpha)$
        Backpropagate $\mathcal{L}_{\text{distill}}$
        Update student model $S$ parameters
    Evaluate the student model $S$ on $D_{\text{val}}$
     Save the student model $S$ if validation accuracy improves
```

Figure 2. Pseudocode of traditional KD model

3.2. Iterative Distillation Process

To maintain the effectiveness of the tutor, the TEID framework includes a mechanism for continuous updating and re-distillation. Specifically, whenever the student model learns from the teacher, the tutor model is also updated to learn from the teacher. This process ensures that the tutor model remains a reliable intermediary for knowledge transfer.



```
Algorithm 2 Tutor-Enhanced Iterative Distillation
Input: Teacher model $T$, Tutor model $U$, Student model $S$, Training data $D_{\text{train}}$, Validation data
       $D_{\text{val}}$, Temperature $\tau$, Weight $\alpha$
Output: Best student model $S$
Initialize the teacher model $T$, tutor model $U$, and student model $S$
 Set the teacher model $T$ and tutor model $U$ to evaluation mode
 for each epoch do
    for each batch $(x, y) \in D_{train}$ do
        $z_T \leftarrow T(x)$                          // Forward pass through teacher
         $z_U \leftarrow U(x)$                           // Forward pass through tutor
         $z_S \leftarrow S(x)$                         // Forward pass through student
         $\mathcal{L}_T \leftarrow \text{DistillationLoss}(z_S, z_T, y, \tau, \alpha)$
         $\mathcal{L}_U \leftarrow \text{DistillationLoss}(z_S, z_U, y, \tau, \alpha)$
         if $\mathcal{L}_T < \mathcal{L}_U$ then
            $\mathcal{L} \leftarrow \mathcal{L}_T$
             Update tutor model $U$ from teacher model $T$
        else
            $\mathcal{L} \leftarrow \mathcal{L}_U$
        Backpropagate $\mathcal{L}$  Update student model $S$ parameters
    Evaluate the student model $S$ on $D_{\text{val}}$
     Save the student model $S$ if validation accuracy improves
```

Figure 3. Pseudo-code of TEID model

To achieve further compression and efficiency, the TEID framework employs an iterative distillation procedure (cf. Figure 2). This process involves repeating the distillation on the tutor model and the resultant student model, using a new, smaller student model at each iteration. The iterative procedure is as follows: i) In the first iteration, the student model learns from both the teacher and the tutor models, and the tutor model is updated as needed, ii) In subsequent iterations, the tutor becomes the new teacher, the previous student model becomes the new tutor, and a new, smaller student model is introduced - the process of selective learning and continuous updating is repeated, iii) This iterative approach enables the creation of progressively smaller and more efficient models while retaining the benefits of the original large-scale teacher model.

4. Evaluation Methodology

This section details the experimental setup used in our study, which aims to evaluate the effectiveness of Tutor-Enhanced Iterative Distillation (TEID) compared to traditional Knowledge Distillation (KD, cf. Figure 4).

4.1. Dataset

We utilize the Stanford Sentiment Treebank (SST-2) dataset from the General Language Understanding Evaluation (GLUE) benchmark. SST-2 is a binary classification task where the goal is to determine the sentiment (positive or negative) of a given sentence. The dataset consists of 67,349 training samples, 872 validation samples, and 1,821 test samples. This dataset is suitable for evaluating the performance of sentiment analysis models.

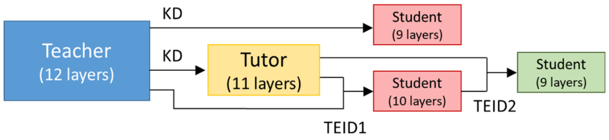


Figure 4. Overview of TEID experiment

4.2. Teacher Model Fine-tuning

The teacher is a pre-trained BERT base model with 12 layers. It is fine-tuned on the SST-2 dataset to adapt it for binary classification. The fine-tuning process involves adjusting the BERT model's final layer to output two logits corresponding to the labels. The training procedure includes: i) Optimizer: AdamW, ii) Learning Rate: 2e-5, iii) Batch Size: 32 for training, 8 for validation, and iv) Number of Epochs: 4.

4.3. Traditional Knowledge Distillation

We consider two models: i) Student model with 9 layers and ii) Tutor model with 11 layers.

Student Model with 9 Layers: A student model with 9 layers is distilled from the fine-tuned 12-layer BERT teacher using traditional KD. The distillation process involves the following steps: i) Compute the logits from the teacher model for each training sample, ii) Train the student model to match the teacher's logits using a combination of cross-entropy loss with the ground truth labels and Kullback-Leibler (KL) divergence loss with the teacher's softened logits, iii) Use a temperature scaling factor of $\tau = 1.0$ and an interpolation weight $\alpha = 0.5$ to balance the two loss components, iv) Apply early stopping with a patience of 2 epochs based on validation accuracy to prevent overfitting.

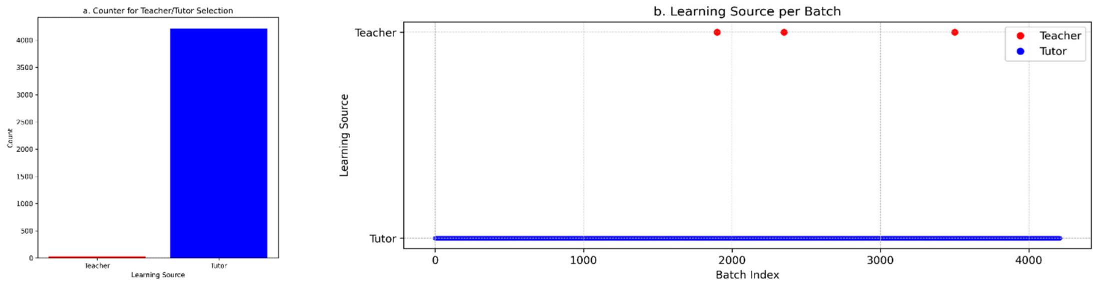
Tutor Model with 11 Layers: tutor model with 11 layers is distilled from the fine-tuned 12-layer BERT teacher using the same KD process described above. This tutor model will later serve as an intermediate model in the TEID process.

4.4. Tutor-Enhanced Iterative Distillation (TEID)

Here, we also consider two models: i) First iteration: 10-layer student, and ii) Second iteration: 9-layer student.

First Iteration: 10-Layer Student: The first iteration of TEID involves distilling knowledge from both the 12-layer teacher and the 11-layer tutor to a 10-layer student model. The process includes: i) Forward pass through the teacher, tutor, and student models for each training batch, ii) Compute the distillation losses between the student and both the teacher and tutor, iii) Select the lower loss and update the student model accordingly, iv) Periodically update the tutor model using the teacher model to ensure it remains a reliable intermediate, and v) Apply early stopping with a patience of 5 epochs based on validation accuracy to ensure training efficiency and prevent overfitting.

Second Iteration: 9-Layer Student: In the second iteration of TEID, the previously trained 11-layer tutor becomes the new teacher, the 10-layer student becomes the new tutor, and a new 9-layer student is introduced.



a. Cumulative count of batches in which the teacher (red) or tutor (blue) was chosen as the supervision source

b. Batch-wise timeline showing, for every training batch, whether the student learned from the teacher (red markers) or the tutor (blue markers).

Figure 5. Teacher/Tutor selection results

5. Empirical Results

This section presents the results of our experiments, comparing the performance of the teacher model, the traditional KD models, and the TEID models on the SST-2 binary classification task. The teacher, a fine-tuned BERT base with 12 layers, achieved an accuracy of 93% on the SST-2 validation set, which is consistent with the scores reported in the literature for BERT base models on the SST-2 dataset. The final 9-layer student obtained from TEID is compared with the other 9-layer student model, previously distilled directly from the 12-layer BERT teacher using traditional KD. Both models are evaluated on the SST-2 validation set using i) accuracy, ii) precision, iii) recall, and iv) F1 score.

5.1. Traditional Knowledge Distillation (KD)

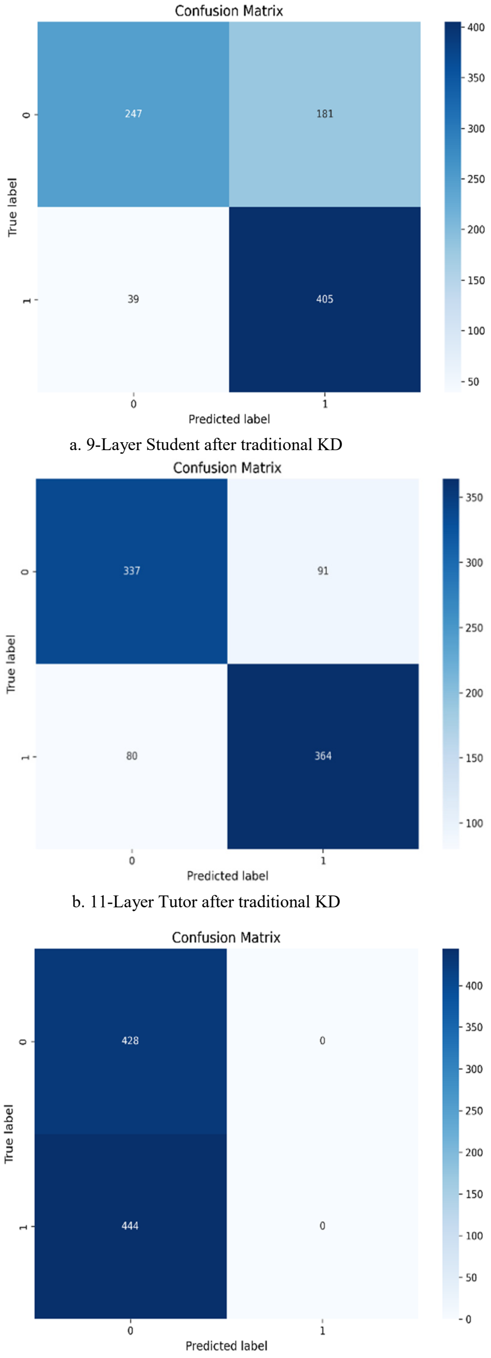
9-Layer Student Model: distilled from the 12-layer teacher using traditional KD achieved a validation accuracy of 74.77%. This indicates a significant drop in performance compared to the teacher, but it demonstrates the feasibility of distilling knowledge into a smaller model (cf. Figure 6.a).

11-Layer Tutor Model: distilled from the 12-layer teacher using traditional KD achieved a validation accuracy of 80.39%. This model served as an intermediate in our TEID approach, showing better performance compared to the 9-layer student model (cf. Figure 6.b).

5.2. Tutor-Enhanced Iterative Distillation (TEID)

10-Layer Student Model: In the first iteration of TEID, we distilled knowledge from both the 12-layer teacher model and the 11-layer tutor model into a 10-layer student model. Unfortunately, the 10-layer student model achieved a validation accuracy of only 49.08%, indicating poor performance (cf. Figure 6.c). Due to this suboptimal result, we decided to stop the TEID process at this iteration. During this iteration, we logged the number of times the teacher model and the tutor model were selected for distillation. The teacher model was selected only 3 times out of 4210 batches, while the tutor model was selected for the remaining batches. This log helps us analyze the effectiveness of selecting the tutor model over the teacher model during the distillation process.

Table 1 summarizes the validation accuracy of all models. The results indicate that while traditional KD can effectively distill knowledge into smaller models, the first iteration of TEID did not perform as well. Further analysis is required to understand the reasons behind the performance of the 10-layer student model in the TEID process.



a. 9-Layer Student after traditional KD

b. 11-Layer Tutor after traditional KD

c. 10-Layer Student after TEID

Figure 6. Tutor-Enhanced iterative distillation results

Table 1. Validation accuracy of various models on the SST-2 dataset

Model	Teacher Model (12 layers)	Student Model (9 layers, KD)	Tutor Model (11 layers, KD)	Student Model (10 layers, TEID)
Validation Accuracy (%)	93.00	74.77	80.39	49.08

The experimental results reveal several critical insights into the performance and limitations of the Tutor-Enhanced Iterative Distillation (TEID) method compared to traditional Knowledge Distillation (KD).

Poor Performance of TEID: The first iteration of TEID, which aimed to distill knowledge from both the 12-layer teacher and the 11-layer tutor into a 10-layer student, resulted in a validation accuracy of only 49.08%. This poor performance prompted us to terminate the TEID process at this stage. The drop in accuracy indicates that the TEID method in this experiment, was not effective in transferring knowledge to the 10-layer student model.

Analysis of Confusion Matrix: The confusion matrix for the 10-layer student model (cf. Figure 6.c) indicates that the model is predicting all instances as class 0 and not predicting any instances as class 1. This behavior results in 100% Recall for class 0 and 0% Precision for class as well as an undefined F-score for class 1 due to the absence of predicted instances, which significantly impacts the overall performance metrics. This imbalance in predictions highlights a severe deficiency in the student model's ability to generalize and correctly classify both classes.

5.3. Tutor Model Selection in TEID

During the TEID process, the selection logs reveal that the tutor was selected almost exclusively, with the teacher being chosen only 3 times out of 4,210 batches (cf. Figure 5). This overwhelming preference for the tutor defies the primary purpose of TEID, which is to leverage both the teacher and tutor to enhance knowledge transfer to the student.

Several factors may contribute to this issue: i) *Loss Comparison Bias*: The distillation loss comparison might inherently favor the tutor model, especially if the tutor's intermediate representations are closer to those of the student, resulting in lower distillation losses, ii) *Suboptimal Tutor Model*: The 11-layer tutor, although better than the student, may not be significantly better than the teacher, leading to an ineffective distillation process, iii) *Implementation Issues*: Potential bugs or biases in the implementation of the TEID process might lead to the tutor being unfairly favored during selection. This behavior suggests that the current TEID implementation needs to be refined to ensure a balanced and effective utilization of both the teacher and tutor models.

6. Conclusion

This paper introduces a novel Tutor-Enhanced Iterative knowledge Distillation (TEID) solution to fill the knowledge capacity gap between LLMs. It innovates over classical KD methods by adding an intermediate-sized tutor model that assists in improved knowledge transfer. TEID uses a selective learning strategy to enable the student model to learn from either the teacher or the tutor model, alongside a continuous updating and re-distillation of the tutor. To achieve further compression, the TEID is repeated iteratively on the tutor and the previously resultant student, with a new smaller student model. Empirical results demonstrate that 12-layer BERT teacher model achieved improved accuracy. Traditional KD yielded a 9-layer student with lesser performance, but the first iteration of TEID showed suboptimal results, with the 10-layer student model performing poorly and consistently predicting a single class. While TEID presents a promising approach to enhance knowledge transfer, our results indicate that significant refinements are needed to achieve its potential. Future work should focus on improving the algorithm to balance the use of teacher and tutor models effectively. Future directions include assessing TEID performance and re-evaluating the loss function to ensure a more balanced comparison between the teacher and tutor, potentially by introducing weights or scaling factors. Another future direction is refining the algorithm to prevent biases in model selection and ensure fair utilization of both teacher and tutor [3, 8].

References

[1] Brown N., et al., *Efficient Transformer Knowledge Distillation: A Performance Review.* Confer. on Empirical Methods in Natural Language Processing (EMNLP) Industry Track, 2023. 54-65.

[2] Elsken T., et al., *Meta-Learning of Neural Architectures for Few-Shot Learning.* IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020. 12362-12372.

[3] Ganta D., et al., *Knowledge Distillation via Weighted Ensemble of Teaching Assistants.* IEEE International Conf. on Knowledge Graph (ICKG'21), 2021. pp. 30-37.

[4] Gao M., et al., *Residual error based knowledge distillation.* Neurocomputing, 2021. 433: 154-161.

[5] Koroteev M., *BERT: A Review of Applications in Natural Language Processing and Understanding.* CoRR abs/2103.11943, 2021.

[6] Li J., et al., *Distilling ChatGPT for Explainable Automated Student Answer Assessment.* Conf. on Empirical Methods in Natural Language Processing (EMNLP), 2023. 6007-6026.

[7] Li L., et al., *Dynamic Knowledge Distillation for Pre-trained Language Models.* Conference on Empirical Methods in Natural Language Processing (EMNLP), 2021. (1): 379-389.

[8] Li Y., et al., *Self-Distillation with Meta Learning for Knowledge Graph Completion.* Findings of the Association for Computational Linguistics: EMNLP, 2022. 2048–2054.

[9] Li Z., et al., *Hint-Based Training for Non-Autoregressive Machine Translation.* Conf. on Empirical Methods in Natural Language Processing (EMNLP), 2019. (1): 5707-5712.

[10] Liu J., et al., *Graph-based Knowledge Distillation: A survey and experimental evaluation.* CoRR abs/2302.14643, 2023.

[11] Liu Y., et al., *Reducing Capacity Gap in Knowledge Distillation with Review Mechanism for Crowd Counting.* CoRR abs/2206.05475 2022.

[12] Mirzadeh S., et al., *Improved Knowledge Distillation via Teacher Assistant: Bridging the Gap Between Student and Teacher.* CoRR abs/1902.03393, 2019.

[13] Moslemi A., et al., *A survey on knowledge distillation: Recent advancements.* Machine Learning with Applications, 2024. 100605.

[14] Ramesh K., et al., *A Comparative Study on the Impact of Model Compression Techniques on Fairness in Language Models.* Annual Meeting of the Association for Computational Linguistics (ACL), 2023. (1): 15762-15782.

[15] Tomut A. et al., *CompactifAI: Extreme Compression of Large Language Models using Quantum-Inspired Tensor Networks.* CoRR abs/2401.14109, 2024.

[16] Wang W. et al., *Model Compression and Efficient Inference for Large Language Models: A Survey.* CoRR abs/2402.09748, 2024.

[17] Zhu X., et al., *A Survey on Model Compression for Large Language Models.* Transactions of the Association for Computational Linguistics, 2024. 12: 1556-1577.

[18] Zou A., et al., *Dynamic multi teacher knowledge distillation for semantic parsing in KBQA.* Expert Systems with Applications, 2025. 263: 125599.