Self-Regulated Data-Free Knowledge Amalgamation for Text Classification

Prashanth Vijayaraghavan
IBM Research
San Jose CA 95120
prashanthv@ibm.com

Hongzhi Wang
IBM Research
San Jose CA 95120
hongzhiw@us.ibm.com

Luyao Shi
IBM Research
San Jose CA 95120
luyao.shi@ibm.com

Tyler Baldwin
IBM Research
San Jose CA 95120
tbaldwin@us.ibm.com

David Beymer
IBM Research
San Jose CA 95120
beymer@us.ibm.com

Ehsan Degan
IBM Research
San Jose CA 95120
edehgha@us.ibm.com

Abstract

Recently, there has been a growing availability of pre-trained text models on various model repositories. These models greatly reduce the cost of training new models from scratch as they can be fine-tuned for specific tasks or trained on large datasets. However, these datasets may not be publicly accessible due to the privacy, security, or intellectual property issues. In this paper, we aim to develop a lightweight student network that can learn from multiple teacher models without accessing their original training data. Hence, we investigate Data-Free Knowledge Amalgamation (DFKA), a knowledge-transfer task that combines insights from multiple pre-trained teacher models and transfers them effectively to a compact student network. To accomplish this, we propose STRATANET, a modeling framework comprising: (a) a steerable data generator that produces text data tailored to each teacher and (b) an amalgamation module that implements a self-regulative strategy using confidence estimates from the teachers' different layers to selectively integrate their knowledge and train a versatile student. We evaluate our method on three benchmark text classification datasets with varying labels or domains. Empirically, we demonstrate that the student model learned using our STRATANET outperforms several baselines significantly under data-driven and data-free constraints.

1 Introduction

Recent NLP advancements have yielded numerous pre-trained models, often achieving state-of-the-art performance across various tasks. These models are publicly available to promote reproducibility and further research. To facilitate knowledge transfer from pre-trained teacher models, Hinton et al. (2015) pioneered Knowledge Distillation (KD), utilizing soft target labels to train light-weight student models effectively. Subsequently, diverse KD approaches have been successfully applied in different domains. Traditionally, KD relies on using original training data to guide the student model's learning from a task-specific teacher model. However, this approach has limitations, often involving learning from a single teacher model (Sanh et al., 2019; Liu et al., 2020) or a task-specific ensemble of teachers (Fukuda et al., 2017; Tian et al., 2019).

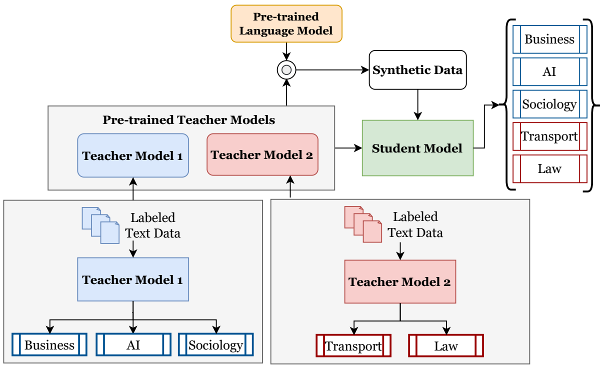

Figure 1: Given a set of pre-trained teacher models (Teacher Models 1 & 2), each with distinct expertise, the goal is to train a student model capable of amalgamating their knowledge, mastering prediction across all specialized classes of the teachers.

Unlike traditional KD, where teachers focus on the same task, knowledge amalgamation (KA) techniques (Luo et al., 2019; Shen et al., 2019) enable learning in a student network by integrating knowledge from multiple teachers with diverse expertise. These methods enhance the student model's classification abilities across a wider range of labels. While KA techniques are well-established in Computer Vision, their exploration in NLP literature is limited. Li et al. (2022) utilized Monte-Carlo Dropout to estimate model uncertainty for merging knowledge from different pre-trained teacher models. However, these techniques often require access to unlabeled data from the original training set used by the pre-trained models (Luo et al., 2019; Shen et al., 2019; Li et al., 2021; Vongkulbhisal et al.,

Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 6: Industry Track), pages 491–502
June 16-21, 2024 ©2024 Association for Computational Linguistics

2019) to train a versatile student model. Unfortunately, the original training data and annotations are often unavailable due to various issues. Moreover, the diverse expertise of teacher models may lead to uncertain states and probabilities when handling input sequences outside their domains. These challenges hinder the application of KA methods in broader domains. To address this, we explore a practical knowledge-transfer task called Data-Free Knowledge Amalgamation (DFKA). Figure 1 provides an overview of this task, aiming to enhance the student model's capabilities by integrating knowledge from multiple pre-trained teachers without access to the original training data.

To achieve our goal, we introduce STRATANET[1], a knowledge amalgamation framework with: (i) a flexible generation module creating pseudo text data for each pre-trained teacher network, and (ii) an amalgamation module enabling self-regulated integration of teachers' knowledge during student model training. Integration is guided by a teacher-specific out-of-distribution (OOD) score, assessing the reliability of intermediate and output states of every pre-trained teacher model.

Contributions: (1) Introduction of STRATANET, a pioneering data-free knowledge amalgamation (DFKA) method for lightweight student model training without accessing original training data. (2) Proposal of a block-wise amalgamation strategy for integrating knowledge from multiple heterogeneous (or homogeneous) teacher model layers into the student model. (3) Demonstration of superior performance by our STRATANET-trained student model compared to various baselines across three benchmark text datasets: AG News, OhSumed Abstracts, and 5 Abstracts Group.

2 Related Work

In this section, we explore the relevant literature concerning knowledge distillation (KD) and amalgamation. KD is a technique aimed at transferring knowledge from a large teacher network to a student model, offering benefits across various NLP tasks and facilitating model compression. These tasks encompass question answering (Izacard and Grave, 2020; Yang et al., 2020), multi-modal summarization (Zhang et al., 2022), and neural machine translation (Tan et al., 2019; Wang et al., 2021; Zhou et al., 2019), among others. Notable approaches such as DistilBERT (Sanh et al., 2019) and TinyBERT (Jiao et al., 2019) primarily focus on compressing models, maintaining the student architecture identical to that of the teacher model (i.e., homogeneous setting). Fewer models, like those by Tang et al. (Tang et al., 2019b,a), train a heterogeneous student model. While KD has found widespread application in NLP, data-free knowledge distillation (DFKD) remains relatively underexplored compared to its application in computer vision. Recent studies (Melas-Kyriazi et al., 2020; Ma et al., 2020, 2022) have delved into training compressed student models under data-free settings using techniques such as training data augmentation, plug & play embedding guessing, and reinforced topic prompter.

In contrast to the singular teacher model approach in KD, knowledge amalgamation (KA) involves training a versatile student model by amalgamating insights from multiple pre-trained teacher models. Li et al. (2022) utilized Monte Carlo Dropout to estimate model uncertainty and perform classification on the union of label sets from different teacher models. Although these methods do not rely on human-annotated labels, they leverage input text from the original training data. Jin et al. (2022) proposed a parameter space merging method for dataless knowledge fusion, assuming an impractical uniformity in model architectures across input and merged models. Differing from the aforementioned approaches, our method, StrataNet, introduces a framework for data-free knowledge amalgamation (DFKA) in text, representing a pioneering exploration in NLP literature involving multiple heterogeneous teacher networks.

3 Problem Setup

Given K pre-trained teacher models $\mathcal{T} = \{\mathcal{T}_i\}_{i=1}^{K}$, each with $L_{\mathcal{T}_i}$-layers and its own domain of expertise, i.e., performing a c_i-class classification task with few overlapping or disjoint set of labels $\mathcal{Y}_i = \{y_i^j\}_{j=1}^{c_i}$, our goal is to train a lightweight student model $\mathcal{S}$ with $L_{\mathcal{S}}$-layers such that it can compute predictions over the union of all the label sets, $\mathcal{Y} = \bigcup_{i=1}^{K} \mathcal{Y}_i$ and $L_{\mathcal{S}} \leq \min(\{L_{\mathcal{T}_i}\}_{i=1}^{K})$.

4 Proposed Approach

4.1 Overview

In this section, we outline our framework, STRATANET, designed to train a lightweight student model using multiple teachers under data-free

[1] Short for **S**elective **T**ransformer based Self-**R**egul**AT**ive **A**malgamation **NET**work

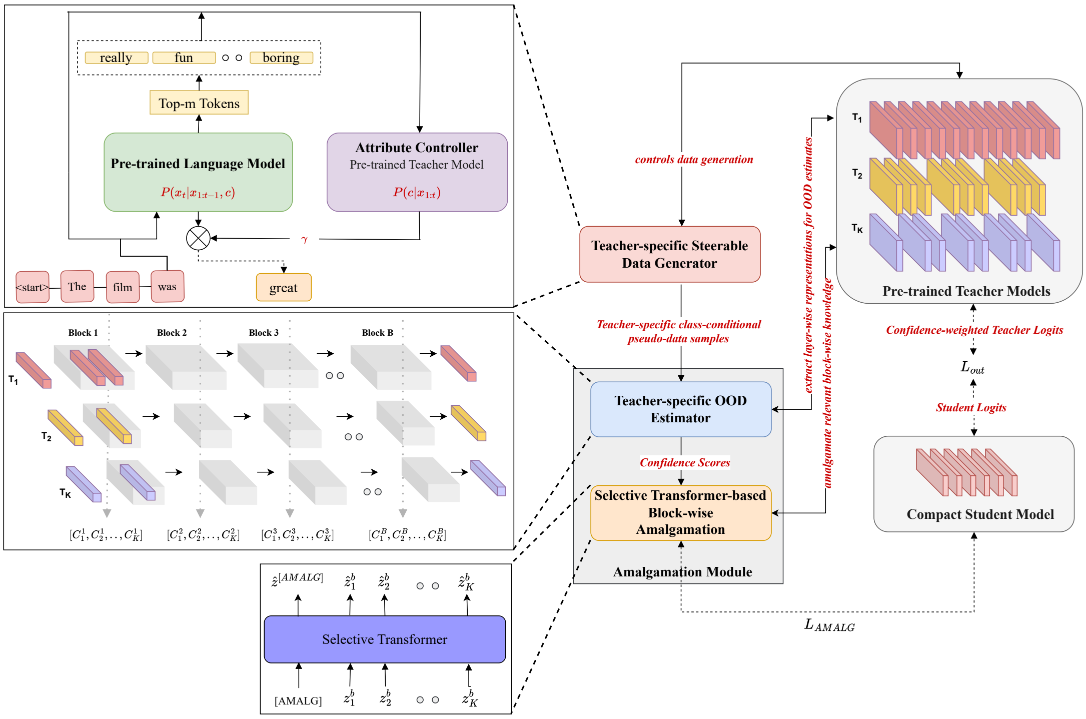
really
fun
boring
Top-m Tokens
Pre-trained Language Model
$P(x_t|x_{1:t-1}, c)$
Attribute Controller
Pre-trained Teacher Model
$P(c|x_{1:t})$
γ
<start>
The
film
was
great
Block 1
Block 2
Block 3
Block B
T1
T2
TK
$[C_1^1, C_2^1, ..., C_K^1]$
$[C_1^2, C_2^2, ..., C_K^2]$
$[C_1^3, C_2^3, ..., C_K^3]$
$[C_1^B, C_2^B, ..., C_K^B]$
$\hat{z}^{[AMALG]}$
$\hat{z}_1^b$
$\hat{z}_2^b$
$\hat{z}_K^b$
Selective Transformer
[AMALG]
z_1^b
z_2^b
z_K^b
controls data generation
Teacher-specific Steerable Data Generator
Teacher-specific class-conditional pseudo-data samples
Teacher-specific OOD Estimator
Confidence Scores
Selective Transformer-based Block-wise Amalgamation
Amalgamation Module
extract layer-wise representations for OOD estimates
amalgamate relevant block-wise knowledge
Pre-trained Teacher Models
Confidence-weighted Teacher Logits
L_{out}
Student Logits
Compact Student Model
L_{AMALG}

Figure 2: Illustration of our STRATANET framework.

constraints. We address the following factors: (a) lack of training data, (b) existence of specialized teachers with non-overlapping or partially overlapping label sets, and (c) need to integrate knowledge from diverse teachers. Our STRATANET consists of two main components. The first, $\mathcal{G}_i$, is a teacher-specific steerable data generator. It guides a base pre-trained language model, $\mathcal{P}$, to generate tailored text for each teacher, $\mathcal{T}_i$, overcoming data scarcity by creating pseudo-data samples. The second component, the amalgamation module, serves two functions. It evaluates each teacher's confidence in predicting within their expertise and employs block-wise integration with a selective transformer to fuse knowledge from multiple teachers. Utilizing confidence scores, this approach appropriately weights representations from different teacher models, effectively managing diverse teacher architectures.

4.2 Steerable Data Generator

To overcome the challenge of unavailability of the original training data for teacher models, we utilize a conditional text generation method that generates pseudo-data samples specifically tailored to the label set of the teacher $\mathcal{T}_i$. Given a teacher model $\mathcal{T}_i$ and any class label $c \in \mathcal{Y}_i$, a steerable text generator, $\mathcal{G}_i$, produces a class-controlled text x of length N as follows: $P(x_{1:N}|c) = \prod_{t=1}^{N} P(x_t|x_{1:t-1}, c)$

For each teacher $\mathcal{T}_i$, our steerable text generator produces pseudo-data samples $\mathcal{D}_i^p = (\mathcal{X}_i^p, \hat{\mathcal{Y}}_i^p)$ by applying an inference-time controllable generation method to steer an unconditional language model towards the desired class label relevant to a specific teacher. The generation process entails guiding a base pre-trained language model (PLM), denoted as $\mathcal{P}$, using a post-processing module. By adjusting the parameters during the decoding phase, the generator exhibits varying degrees of class control over the text sampled from the chosen base PLM.

Based on a recent study by Gu et al. (2022), we adopt a variant of the weighted decoding method to generate class-conditional text using a pre-trained unconditional language model, denoted as $\mathcal{P}$. In this approach, we model the generation process by incorporating a Bayesian factorization as follows:

$$P(x_t|x_{1:t-1}, c) \propto P(x_t|x_{1:t-1}) P(c|x_{1:t})^{\gamma} \quad (1)$$

Here, γ represents a hyperparameter for control strength. The first term corresponds to the output probabilities generated by the chosen PLM, while the second term relies on the teacher model to estimate the likelihood of the generated text (up to the current time step t) being classified under the class label c. During the sampling process, the value of γ regulates the influence of the teacher model.

One challenge in this approach is the computational complexity of teacher-guided sequence sampling. To compute the second term in Equation 1, we need to estimate the class probability $P(c|x_{1:t})$, requiring evaluation of $P(c|x_{1:t-1}, x_t)$ for every token in the vocabulary $\mathcal{V}$ at the t^{th} timestep. To reduce inference time, we exclude low-probability tokens and prioritize a subset for teacher guidance. Tokens with low probability $P(x_t|x_{1:t-1})$ from the PLM are discarded, even if the teacher model assigns high weights $P(c|x_{1:t})$. Consequently, we exclusively use the top-m tokens with higher probabilities, guided by the teacher model's weights. Subsequently, we employ a top-k sampling strategy, where $k < m \ll |\mathcal{V}|$). Our experiments indicate that setting $m = 100$ is notably effective. Table 1 displays sample generations produced using the teacher-guided generation module. In this experiment, we trained two teachers on the AG News and OhSumed label sets. Subsequently, we generated pseudo-data samples for the "Sports" category under the guidance of the teacher trained on the AG News dataset and for the "Cardiovascular Diseases" category under the guidance of the teacher trained on the OhSumed dataset. Notably, we observe that the generated text is not only fluent but also relevant to the respective category of interest.

Dataset	Text Generated
AG News	**Category: Sports** In an electrifying moment that left spectators spellbound, Olympic speedster Usain Bolt once again proved that he is the fastest man alive by shattering yet another world record. As he crossed the finish line, the Jamaican sprinter scorched towards the tape while leaving all competition behind ... In front of thousands of cheering fans, Bolt completed the race faster than his previous mark set last year. It was a performance that brought tears of joy to the eyes....
OhSumed	**Category: Cardiovascular Diseases** The study aimed to determine the prevalence of echocardiographic aortic regurgitation among patients presenting for screening echocardiography at a single university center. Echocardiograms were performed in accordance with Echocardiography guidelines ... New findings reveals important data addressing our knowledge gap regarding Aortic Regurgitation patient prevalence. In this study, transthoracic imaging confirmed prevalence across all ages stratified by 10yr increments and between men and women. The report sheds light on the epidemiology of AR found by echocardiogram ....

Table 1: Sample Generations from our Steerable Generation Module

4.3 Block-wise Amalgamation Module

We introduce an amalgamation module that estimates out-of-distribution (OOD) scores for each teacher using a Teacher-specific OOD Estimator. It integrates selective informative states from relevant teachers based on these scores using a Selective Transformer (ST-AMALG), transferring them to a student in a blockwise manner to accommodate varying sizes of teacher models.

4.4 Teacher-specific OOD Estimator

Due to diverse label sets $\{\mathcal{Y}_i\}_{i=1}^{K}$ in pre-trained teacher models $\{\mathcal{T}\}_{i=1}^{K}$, any input text from an unseen category for a specific teacher is considered out-of-distribution (OOD). Extracted features from that teacher's intermediate layer may not be sufficient for effective knowledge transfer to the student model. Studies indicate: (a) Transformer-based models encode transferable features in various intermediate layers (Liu et al., 2019; Rogers et al., 2021), and (b) final layers, especially in models like BERT, are highly task-specific (Kovaleva et al., 2019; Rogers et al., 2021). Considering these, we propose layer-wise teacher-specific lightweight OOD estimators, explained below.

4.4.1 OOD Score Computation

For an input text $x \in \mathcal{X}_i$ with label $y \in \hat{\mathcal{Y}}_i$, a transformer-based pre-trained teacher model $\mathcal{T}_i$ produces contextual token-level latent embeddings at each layer $l \in L_{\mathcal{T}_i}$. These are averaged into a single latent representation $h_i^l \in \mathcal{R}^{d_i}$, where d_i is the dimensions of the latent representations. To compute an OOD score for any new input x_{new}, we use a Mahalanobis distance (MD) based OOD detection technique. For an in-distribution (ID) dataset with c_i-labels associated with $\mathcal{T}_i$, the MD technique fits c_i-class conditional Gaussian distributions $\mathcal{N}(\mu_y, \Sigma)$ to each of the c_i ID classes based on training latent representations h_i^l. However, Ren et al. (2021) proposed a Relative Mahalanobis distance (RMD) that outperforms MD in OOD detection for both near and far-OOD scenarios by calculating the distance between class-conditional Gaussians and a single background Gaussian using data from all classes. For an input x_{new} with the latent representation $\hat{h}_i^l$ at layer l, RMD is given by:

$$\textsc{Rmd}_y(\hat{h}_i^l) = \textsc{Md}_y(\hat{h}_i^l) - \textsc{Md}_{bg}(\hat{h}_i^l) \quad (2)$$

$$\textsc{Md}_y(\hat{h}_i^l) = (\hat{h}_i^l - \mu_y)^T \Sigma^{-1} (\hat{h}_i^l - \mu_y) \quad (3)$$

$$\mathcal{C}_i^l(\hat{h}_i^l) = -min_y\{\textsc{Rmd}_y(\hat{h}_i^l)\} \quad (4)$$

where $\mathcal{C}_i^l$ refers to the confidence score of x_{new} being in-domain for $\mathcal{T}_i$ based on representation at layer l, μ_y is a class-conditional mean vectors and Σ is the covariance matrix, MD_{bg} indicates Mahalanobis distance of h_i^l to the background distribution fitted to the entire training data usually. The RMD score acts as a contrastive measure indicating the sample's proximity to both the training and background domains. Higher scores indicate greater out-of-distribution characteristics, resulting in lower ID confidence scores, $\mathcal{C}_i^l$. Alternatively, in some cases, intermediate layers can be partitioned into B blocks. Applying a similar procedure as described in Equations (2)-(4), confidence scores can be calculated for each block. Here, $\hat{h}_i^b$ represents the latent representation for block b, obtained by mean pooling over the layer representations within that block. We use a held-out subset of pseudo-data samples, $\hat{\mathcal{D}}_i^p$, generated for each teacher $\mathcal{T}_i$.

4.5 Selective Transformer-based Block-wise Amalgamation ST-AMALG

To transfer knowledge from diverse, larger teachers to a lightweight student model, we align intermediate representations in a block-wise manner, accommodating the varying number of layers between them. Each teacher network $\mathcal{T}_i$ may have a different number of grouped layers. We compute confidence-aware block-wise intermediate representations, z_i^b, using the confidence score at each block b for each teacher. Inspired by the literature on multimodal analysis (Urooj et al., 2020; Vijayaraghavan and Roy, 2023; Lin et al., 2022), we consider the intermediate latent vectors from K teachers, denoted as $\{z_i^b\}_{i=1}^K$, as a token sequence fed into a Transformer layer. We introduce a learnable special token $[AMALG]$, similar to $[CLS]$, to integrate confidence-enriched representations from teachers into a final block-level amalgamated representation, denoted as $\hat{z}_{\mathcal{T}}^b$. Therefore, we refer to this layer as the Selective Transformer-based amalgamation layer (ST-AMALG). Formally,

$$z_i^b = f(h_i^b) + g(\mathcal{C}_i^b) \tag{5}$$

$$\hat{z}_{\mathcal{T}}^b = \text{ST-AMALG}(\{z_i^b\}_{i=1}^K) \tag{6}$$

where f, g are linear layers to enrich the block-level embeddings.

5 Training Objectives & Details

To amalgamate knowledge at intermediate layers, we compute L2-normalized distance between the student's projected block-level representation and the corresponding teachers' amalgamated embedding. Formally,

$$\begin{aligned} \mathcal{L}_{\text{AMAL}} &= \sum_{b=1}^{B} \mathcal{L}_{\text{AMAL}}^b \\ s.t. \quad \mathcal{L}_{\text{AMAL}}^b &= ||\hat{z}_{\mathcal{S}}^b - \hat{z}_{\mathcal{T}}^b||_2^2 \end{aligned} \tag{7}$$

For the output prediction layer, we compute the KL divergence loss based on confidence weighted combination of Teacher models and the temperature τ as: $\mathcal{L}_{out} = KL(\hat{\mathcal{T}}(x), \hat{\mathcal{S}}, \tau)$.

5.1 Training details

Given steerable data generators $\{\mathcal{G}_i\}_{i=1}^K$ tied to teachers $\{\mathcal{T}_i\}_{i=1}^K$, we produce a student training transfer set, denoted as $\mathcal{D}^p$, by combining the pseudo-data samples generated for all the labels associated with each teacher. Next, we divide the intermediate layers into B-blocks such that the number of layers in each block may vary according to the number of layers in the teacher model. In our experiments, the teacher models (Teacher 1 and Teacher 2) are based on BERT-base-uncased (Devlin et al., 2018), and we set B to the number of intermediate layers in the compressed student model $\mathcal{S}$, i.e., BERT_6. We then compute the number of layers within each block for each teacher accordingly. A subset of pseudo-data samples generated for each teacher $\mathcal{T}_i$, represented as $\hat{\mathcal{D}}_i^p$, is used compute the layer-wise distribution statistics for OOD estimation. Finally, we use the student training transfer set $\mathcal{D}^p$ to train the student model by: (a) computing the confidence of teachers' block-wise features in predicting each input text, (b) amalgamating the confidence-enriched representations from teachers and (c) optimizing the weighted sum of intermediate ($\mathcal{L}_{\text{AMAL}}$) and output prediction layer ($\mathcal{L}_{out}$) losses, expressed as:

$$\mathcal{L} = \lambda \cdot \mathcal{L}_{\text{AMAL}} + (1 - \lambda) \cdot \mathcal{L}_{out} \tag{8}$$

6 Experiments

Our experiments address the following research questions: **(RQ1)** How does our model compare to baseline approaches for knowledge distillation in both data-driven and data-free scenarios? **(RQ2)** What is the individual impact of each component in our model on overall performance? **(RQ3)** How does our model fare when multiple heterogeneous teachers are utilized?

Datasets	#Classes	#Train	#Valid	#Test
AG News	4	108,000	12,000	7,600
5Abstracts Group	5	4,770	530	1,000
OhSumed	23	3,021	336	4,043

Table 2: Data Statistics of benchmark text classification datasets.

6.1 Datasets

We evaluate our approach using the following benchmark datasets: (a) **AG News**[2] (Zhang et al., 2015): It consists of news articles grouped into four major classes—World, Sports, Business, and Sci/Tech. (b) **5 Abstract Group**[3] (Liu et al., 2017): This dataset contains academic paper abstracts from five different domains—business, AI, sociology, transport, and law. (c) **Ohsumed**[4] (Joachims, 1998): It comprises medical abstracts specifically related to cardiovascular diseases. We focus on single-label text categorization and exclude documents that belong to multiple categories. The data statistics for these benchmark datasets are presented in Table 2.

6.2 Baselines

We conduct a comparative analysis of our proposed model with data-driven and data-free baselines. Here is a summary of the baselines:

Teacher Models, which are used to predict individually. We assign zero probabilities to classes outside the expertise of each teacher. **Ensemble**, which concatenates the output logits from all the teachers to obtains predictions over all the labels $\mathcal{Y}$. **MUKA-Hard/Soft** (Li et al., 2021), which is a data-driven KA method that uses Monte-Carlo Dropout based model uncertainty to guide the student training. **Vanilla KA** (Hinton et al., 2015) (R/CD): which aims to mimic the soft targets produced by the logits combination of all teacher models using KL-divergence. In a data-free scenario, we consider two settings: (i) Random Text (R): The student model is trained on text sequences constructed using randomly selected words from the vocabulary of the pre-trained teacher models; and (ii) Cross-Domain Texts (CD): The student model is trained on cross-domain text corpora like Wikitext-103. **As-Dfd** (Ma et al., 2020), which is a data-free knowledge distillation approach. We modify this model for the DFKA scenario by crafting pseudo-embeddings for each teacher as specified in their original study and train a student model using self-supervision and KL-divergence. **StrataNet**, which is our complete DFKA model that generates pseudo-data samples and leverages the produced data for knowledge amalgamation.

Models	AG News	5Abstracts Group	OhSumed
Supervised	94.6	90.7	70.5
Data-Driven Methods			
Teacher 1*	49.9	42.0	36.2
Teacher 2*	47.5	51.5	38.18
Ensemble*	59.8	62.3	45.48
MUKA-Hard*	87.0 (±0.40)	79.0 (±0.82)	—
MUKA-Soft*	87.1 (±0.19)	79.3 (±0.85)	—
Data-Free Methods			
Teacher 1	45.8	41.75	32.8
Teacher 2	46.9	46.88	35.6
Ensemble	55.86	53.67	41.94
Vanilla KA (R)	58.9 (±3.19)	56.27 (±2.76)	47.33 (±4.41)
Vanilla KA (CD)	62.43 (±2.62)	61.55 (±0.91)	50.91 (±2.8)
As-Dfd	74.89 (±0.89)	69.83 (±1.06)	56.08 (±1.6)
StrataNet (Ours)	**88.76** (±0.19)	**83.6** (±0.28)	**65.92** (±0.41)

Table 3: Evaluation results on benchmark text classification dataset averaged over 3 runs. Our method achieve statistically significant improvements over the closest baselines ($p < 0.01$). Bold face indicates the best results and * refers to results from prior literature.

6.3 Metrics

To be comparable with prior studies, we compute the classification accuracy across various datasets. In particular, we report the mean and standard deviations of the accuracy over three runs in §7.

7 Results and Discussion

Overall Performance The evaluation results are presented in Table 3, providing a summary of our findings. To ensure a fair comparison, our baselines incorporate cross-domain data (CD), similar to our model that utilizes a resource like PLM. Additionally, we implement a variation of the data-free knowledge distillation method (Ma et al., 2020) for DFKA. Compared to all the baselines, our StrataNet model demonstrates significant improvement over other DFKA baselines across various text classification datasets. Notably, our com-

[2] `http://groups.di.unipi.it/~gulli/AG_corpus_of_news_articles.html`

[3] `https://github.com/qianliu0708/5AbstractsGroup`

[4] `https://disi.unitn.it/moschitti/corpora.htm`

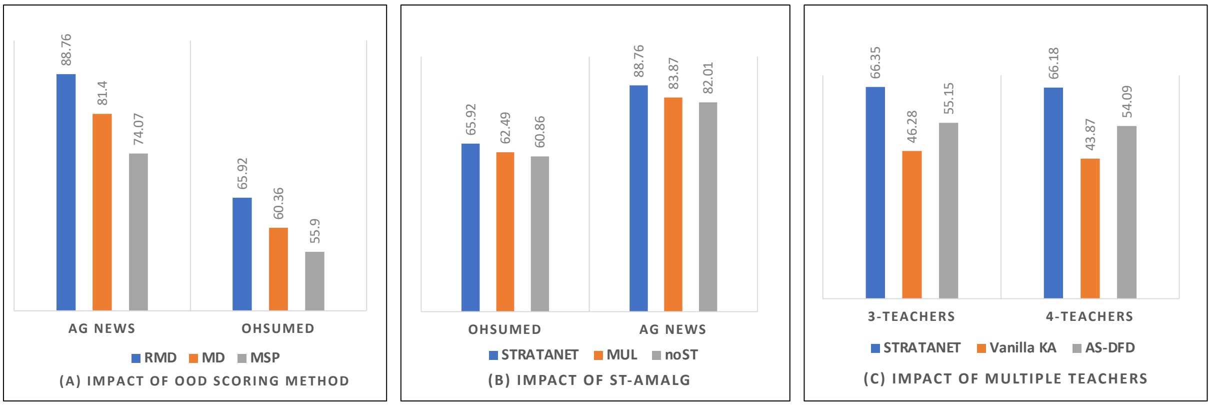

Figure 3: (A) Impact of different OOD scores – RMD, MD & MSP, (B) Impact of ST-AMALG, (C) Effect of Multiple Heterogeneous teachers on OhSumed dataset.

pact student model trained under data-free settings shows an approximately 4% increase in performance compared to the best-performing data-driven model in certain cases. We intuit that the knowledge from the intermediate layers are beneficial for the performance improvement.

7.1 Ablation Studies (RQ2)

7.1.1 Effect of RMD

In order to measure the effect of RMD (explained in §4.4), we replace the OOD score computation using other methods including: (a) embedding-based Mahalanobis distance (MD) and (b) maximum softmax probability (MSP) at the final layer. Figure 3(A) shows how modifying the OOD score has a significant impact on the overall performance of the model. RMD OOD score helps achieve the best performance of our model.

7.1.2 Impact of ST-AMALG

To evaluate the contribution of ST-AMALG, we introduce two variants: (a) STRATANET$_{mul}$:simply multiply the block-level confidence score with the teacher embeddings instead of the embedding enrichment (as in Equation 5, (b) STRATANET$_{noST}$: remove ST-AMALG and use a linear layer on top of confidence weighted sum of teachers' latent vectors in Equation 6. Figure 3(B) shows that both the variants lead to significant performance degradation, asserting their value to the overall model performance. This validates our intuition that the embedding enrichment and ST-AMALG serve as critical components to select the important block-level features from different teacher models [5].

7.2 Effect of Multiple Heterogeneous Teachers (RQ3)

To demonstrate our model's ability to generalize across multiple heterogeneous teachers, we explore scenarios with three (1 BERT-base, 1 RoBerta-base, and 1 ALBERT) and four (1 BERT-base, 2 RoBerta-base, and 1 ALBERT) teachers, each with different architectures. Results are shown in Figure 3(C). While baseline KA methods struggle with increased teacher diversity, our approach consistently improves accuracy and maintains performance with more teachers. These findings underscore the robustness and effectiveness of our method across diverse experimental setups.

8 Conclusion

In this study, we introduce Data-Free Knowledge Amalgamation (DFKA), a method to train a lightweight student network from diverse teacher models without their original training data. Our framework, STRATANET, employs a steerable data generator and an amalgamation module for effective knowledge transfer. Experimental results on text datasets demonstrate the superiority of STRATANET over various baselines, both in data-driven and data-free scenarios. Ablation studies highlight the importance of different model components. This work opens avenues for efficient knowledge transfer in text classification, offering practical solutions for resource-constrained environments.

Limitations

While our STRATANET model outperforms existing baselines, it has certain limitations. The steerable generation module, which guides text genera-

[5] Additional experiments on using LLM like Llama-2 for the data generation module in Appendix B.2

tion for specific classes, may not consistently produce accurate class-specific text. Moreover, it may not capture the full diversity of complex training datasets. Further research is needed to investigate and improve the generation module. Additionally, there is potential to expand knowledge amalgamation to tasks beyond text classification, which warrants future research.

Ethics Statement

Our STRATANET model focuses on improving the performance of DFKA and does not introduce new ethical concerns compared to other KD/KA methods. However, we want to acknowledge two key risks here: (a) data-free knowledge amalgamation strategies can potentially be used as a precursor to model extraction attacks, compromising the confidentiality of blackbox models, as demonstrated in (Truong et al., 2021), and (b) model compression itself may introduce biases, as suggested by (Hooker et al., 2020). It is important to address these risks, which are not specific to our method but are common in data-free model compression techniques, in future research.

References

John Joon Young Chung, Ece Kamar, and Saleema Amershi. 2023. Increasing diversity while maintaining accuracy: Text data generation with large language models and human interventions. *arXiv preprint arXiv:2306.04140*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Takashi Fukuda, Masayuki Suzuki, Gakuto Kurata, Samuel Thomas, Jia Cui, and Bhuvana Ramabhadran. 2017. Efficient knowledge distillation from an ensemble of teachers. In *Interspeech*, pages 3697–3701.

Yuxuan Gu, Xiaocheng Feng, Sicheng Ma, Jiaming Wu, Heng Gong, and Bing Qin. 2022. Improving controllable text generation with position-aware weighted decoding. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3449–3467.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Sara Hooker, Nyalleng Moorosi, Gregory Clark, Samy Bengio, and Emily Denton. 2020. Characterising bias in compressed models. *arXiv preprint arXiv:2010.03058*.

Gautier Izacard and Edouard Grave. 2020. Distilling knowledge from reader to retriever for question answering. *arXiv preprint arXiv:2012.04584*.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2019. Tinybert: Distilling bert for natural language understanding. *arXiv preprint arXiv:1909.10351*.

Xisen Jin, Xiang Ren, Daniel Preotiuc-Pietro, and Pengxiang Cheng. 2022. Dataless knowledge fusion by merging weights of language models. *arXiv preprint arXiv:2212.09849*.

Thorsten Joachims. 1998. Text categorization with support vector machines: Learning with many relevant features. In *European conference on machine learning*, pages 137–142. Springer.

Olga Kovaleva, Alexey Romanov, Anna Rogers, and Anna Rumshisky. 2019. Revealing the dark secrets of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4365–4374, Hong Kong, China. Association for Computational Linguistics.

Lei Li, Yankai Lin, Xuancheng Ren, Guangxiang Zhao, Peng Li, Jie Zhou, and Xu Sun. 2021. Model uncertainty-aware knowledge amalgamation for pre-trained language models. *arXiv preprint arXiv:2112.07327*.

Lei Li, Yankai Lin, Xuancheng Ren, Guangxiang Zhao, Peng Li, Jie Zhou, and Xu Sun. 2022. From mimicking to integrating: Knowledge integration for pre-trained language models. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6391–6402, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Fangjian Lin, Sitong Wu, Yizhe Ma, and Shengwei Tian. 2022. Full-scale selective transformer for semantic segmentation. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, pages 2663–2679.

Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. 2019. Linguistic knowledge and transferability of contextual representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1073–1094, Minneapolis, Minnesota. Association for Computational Linguistics.

Qian Liu, Heyan Huang, Yang Gao, Xiaochi Wei, and Ruiying Geng. 2017. Leveraging pattern associations for word embedding models. In *Database Systems for Advanced Applications: 22nd International Conference, DASFAA 2017, Suzhou, China, March 27-30, 2017, Proceedings, Part I 22*, pages 423–438. Springer.

Weijie Liu, Peng Zhou, Zhiruo Wang, Zhe Zhao, Haotang Deng, and Qi Ju. 2020. FastBERT: a self-distilling BERT with adaptive inference time. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6035–6044, Online. Association for Computational Linguistics.

Renqian Luo, Liai Sun, Yingce Xia, Tao Qin, Sheng Zhang, Hoifung Poon, and Tie-Yan Liu. 2022. Biogpt: generative pre-trained transformer for biomedical text generation and mining. *Briefings in Bioinformatics*, 23(6).

Sihui Luo, Xinchao Wang, Gongfan Fang, Yao Hu, Dapeng Tao, and Mingli Song. 2019. Knowledge amalgamation from heterogeneous networks by common feature learning. *arXiv preprint arXiv:1906.10546*.

Xinyin Ma, Yongliang Shen, Gongfan Fang, Chen Chen, Chenghao Jia, and Weiming Lu. 2020. Adversarial self-supervised data-free distillation for text classification. *arXiv preprint arXiv:2010.04883*.

Xinyin Ma, Xinchao Wang, Gongfan Fang, Yongliang Shen, and Weiming Lu. 2022. Prompting to distill: Boosting data-free knowledge distillation via reinforced prompt. *arXiv preprint arXiv:2205.07523*.

Luke Melas-Kyriazi, George Han, and Celine Liang. 2020. Generation-distillation for efficient natural language understanding in low-data settings. *arXiv preprint arXiv:2002.00733*.

Jie Ren, Stanislav Fort, Jeremiah Liu, Abhijit Guha Roy, Shreyas Padhy, and Balaji Lakshminarayanan. 2021. A simple fix to mahalanobis distance for improving near-ood detection. *arXiv preprint arXiv:2106.09022*.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2021. A primer in bertology: What we know about how bert works. *Transactions of the Association for Computational Linguistics*, 8:842–866.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Chengchao Shen, Xinchao Wang, Jie Song, Li Sun, and Mingli Song. 2019. Amalgamating knowledge towards comprehensive classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3068–3075.

Xu Tan, Yi Ren, Di He, Tao Qin, Zhou Zhao, and Tie-Yan Liu. 2019. Multilingual neural machine translation with knowledge distillation. *arXiv preprint arXiv:1902.10461*.

Raphael Tang, Yao Lu, and Jimmy Lin. 2019a. Natural language generation for effective knowledge distillation. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*, pages 202–208.

Raphael Tang, Yao Lu, Linqing Liu, Lili Mou, Olga Vechtomova, and Jimmy Lin. 2019b. Distilling task-specific knowledge from bert into simple neural networks. *arXiv preprint arXiv:1903.12136*.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. 2019. Contrastive representation distillation. *arXiv preprint arXiv:1910.10699*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Jean-Baptiste Truong, Pratyush Maini, Robert J Walls, and Nicolas Papernot. 2021. Data-free model extraction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4771–4780.

Aisha Urooj, Amir Mazaheri, Niels Da vitoria lobo, and Mubarak Shah. 2020. MMFT-BERT: Multimodal Fusion Transformer with BERT Encodings for Visual Question Answering. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4648–4660, Online. Association for Computational Linguistics.

Prashanth Vijayaraghavan and Deb Roy. 2023. M-sense: Modeling narrative structure in short personal narratives using protagonist's mental representations. *arXiv preprint arXiv:2302.09418*.

Jayakorn Vongkulbhisal, Phongtharin Vinayavekhin, and Marco Visentini-Scarzanella. 2019. Unifying heterogeneous classifiers with distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3175–3184.

Fusheng Wang, Jianhao Yan, Fandong Meng, and Jie Zhou. 2021. Selective knowledge distillation for neural machine translation. *arXiv preprint arXiv:2105.12967*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Ze Yang, Linjun Shou, Ming Gong, Wutao Lin, and Daxin Jiang. 2020. Model compression with two-stage multi-teacher knowledge distillation for web question answering system. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, pages 690–698.

Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *NIPS*.

Zhengkun Zhang, Xiaojun Meng, Yasheng Wang, Xin Jiang, Qun Liu, and Zhenglu Yang. 2022. Unims: A unified framework for multimodal summarization with knowledge distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11757–11764.

Chunting Zhou, Graham Neubig, and Jiatao Gu. 2019. Understanding knowledge distillation in non-autoregressive machine translation. *arXiv preprint arXiv:1911.02727*.

A Implementation Details

We base our STRATANET implementation on PyTorch[6], Huggingface (Wolf et al., 2019) and PyTorch Lightning[7]. We tune our model hyperparameters using grid-search. For the generation module, we sample a maximum of 128 tokens. The top 200 tokens were selected using the nucleus sampling method with a sampling threshold of $p = 0.9$. For Ohsumed dataset, we used BioGPT (Luo et al., 2022) in order to tailor the data generation process to the domain of interest. Trained on large-scale PubMed abstracts, BioGPT is a specialized Transformer language model designed for generating and mining biomedical text. In our experiments, we use a compressed BERT model with 6 layers, referred to as BERT_6, as our student model. Table 4 shows the tuned hyperparameters used by both the generation and distillation component of our STRATANET model. Our method trains a compressed student model (e.g., BERT_6) using a confidence score that selectively amalgamates the knowledge from intermediate and output layers of multiple teachers.

Hyperparameter	Value
Pre-trained LM	GPT-2 (S/M/L) or BioGPT
Learning Rate	2e-5
Batch Size	16
#Epochs	10
Dropout	0.2
Optimizer	AdamW
Learning Rate Scheduling	linear
Weight Decay	0.01
Warmup	2 epochs
Gradient Clipping	1.0
Sampling Method	Nucleus
Sampling - p	0.9
KD Temperature - τ	0.75

Table 4: Hyperparameters used by different components of our proposed PRODGEN model.

B Ablation Studies

B.1 Effect of heterogeneous teachers and student model layers

In Section §6, we conducted experiments using a compressed BERT_6 model, and the results demonstrated no significant performance degradation. To delve deeper, we run additional experiments involving $\{6, 4\}$-layer student models with different teacher configurations: a homogeneous setting ($\mathcal{T}_1, \mathcal{T}_2$: BERT_{large}) and a heterogeneous setting ($\mathcal{T}_1$: BERT_{large}, $\mathcal{T}_2$: RoBERTA_{large}). The evaluations on the AG News dataset reveal the poor performance of the data-free baseline AS-DFD with compressed layers, highlighting the challenges of the heterogeneous setting. However, our STRATANET framework demonstrates consistent and robust performance under both configurations, even with higher compression.

Models	Homogeneous	Heterogeneous
Teacher 1	49.8	48.9
Teacher 2	48.86	50.6
Ensemble	60.25	60.54
AS-DFD_6	75.16	63.89
STRATANET_6	**89.16**	**88.53**
AS-DFD_4	72.80	61.72
STRATANET_4	**88.29**	**86.65**

Table 5: Ablation Study: Effect of heterogeneous teachers & number of student layers

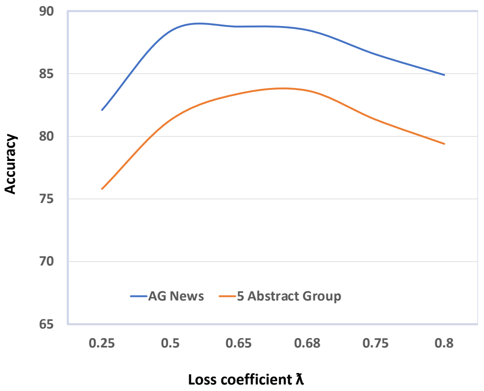

Figure 4: Effect of modifying λ.

Importance of Intermediate Layers: We conduct a sensitivity analysis by varying λ in the loss function, which is associated with the knowledge from intermediate layers. Figure 4 presents the effects of different λ values on the AG News and 5 Abstracts Group datasets. We find that the model performs best with $\lambda \sim 0.65$, indicating the relatively higher importance of intermediate layers for improving performance. This finding aligns with prior studies (Liu et al., 2019; Rogers et al., 2021), which have observed that Transformer-based models often encode transferable features in their intermediate layers.

B.2 Impact of Steerable Data Generation

We evaluate the impact of the Steerable Data generation module through LLM_{Manual}, involving manual prompting of an LLM like Llama-2 (Touvron et al., 2023) using task-specific prompts and em-

[6] https://pytorch.org/

[7] https://www.pytorchlightning.ai/

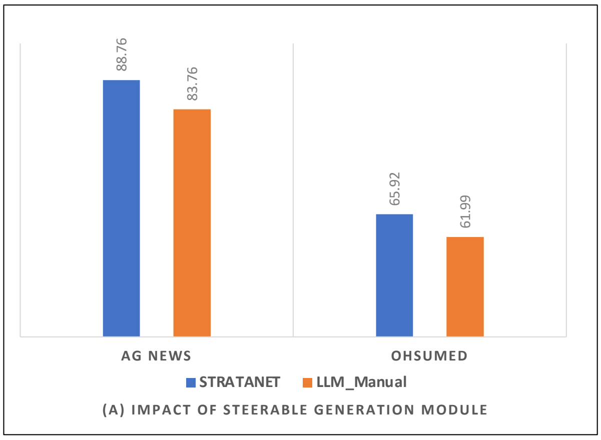

Figure 5: Effect of Steerable Data Generation. Llama-2 with manually designed prompts doesn't outperform our generation module.

Datasets	Manual Prompts
AG News	Generate a [Category] news <article/story>
DBPedia	Generate a document about [Category]
IMDb	Generate a [Category] movie review
SST-2	Generate a [Category] sentence
OhSumed	Generate an abstract about [Category]

Table 6: Samples of dataset-specific manually designed prompts provided as input to the Llama-2 (llama-2-70b-chat) model.

ploying diversification techniques (DTs) like sampling variations and temperature adjustments as described in (Chung et al., 2023). Figure 5 shows no significant performance improvement with a more potent Llama-2 model. While relying solely on manual prompting may lack dataset diversity, diversification techniques enhance performance but might introduce irrelevant tokens, impacting overall generation accuracy. Details of the manually designed prompts are given below.

B.2.1 Manually-designed Prompts

Table 6 show samples of the manually designed prompts to the Llama-2 model.

B.2.2 Generation Parameters

For diversification, we use different temperature setting while we sample tokens. We used five temperature values $\rho \in \{0.3, 0.5, 0.7, 0.9, 1.3\}$. Furthermore, we also experimented with different sampling techniques. For nucleus sampling, we varied the top-p between $\{0.65, 0.95\}$. For top-k sampling, we chose $k \in \{10, 25, 35, 50, 75\}$.