"Call My Big Sibling (CMBS)" – A Confidence-Based Strategy to Combine Small and Large Language Models for Cost-Effective Text Classification

Anonymous ACL submission

Abstract

Transformers have achieved cutting-edge results, with Large Language Models (LLMs) being considered the SOTA in several NLP tasks. However, the literature has not yet fully demonstrated that LLMs are always superior to first-generation Transformers (a.k.a. Small Language Models (SLMs)) in all NLP tasks and scenarios. This study compares three SLMs (BERT, RoBERTa, and BART) with open LLMs (LLaMA 3.1, Mistral, Falcon) across 9 *sentiment analysis* and 4 *topic classification* datasets. The results indicate that open LLMs can moderately outperform or tie with SLMs in all tested datasets, though only when fine-tuned, **at a very high computational cost**. Given this very high cost for only moderate effectiveness gains (3.1% on average), the applicability of these models in practical cost-critical scenarios is questioned. In this context, we propose "Call My Big Sibling" (CMBS)[1], a confidence-based strategy that smoothly combines calibrated SLMs with open LLMs based on prediction certainty. Documents with high (calibrated) confidence are classified by the cheaper SLM, while uncertain documents are directed to LLMs in zero-shot, in-context, or partially-tuned versions. Experiments show that CMBS outperforms SLMs and is very competitive with fully tuned LLMs in terms of effectiveness at a fraction of the latter's cost, offering a much better cost-effectiveness balance.

1 Introduction

Automatic text classification (ATC), such as binary sentiment analysis and topic classification, is essential in diverse contexts, ranging from organizing large data volumes to personalizing user experiences. ATC has experienced a huge revolution with the advent of semantically enriched Transformer models (Devlin et al., 2019) that have achieve state-of-the-art performance in most ATC scenarios (de Andrade et al., 2023; Cunha et al., 2023a; Zanotto et al., 2021; Pasin et al., 2024).

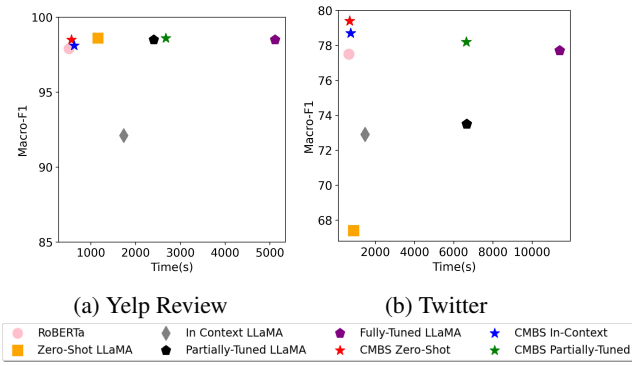


Figure 1: Total Time (seconds) and Macro-F1 in RoBERTa, Zero-Shot LLaMA, In-Context, Partially-Tuned LLaMA, Fully-Tuned LLaMA, CMBS Zero-Shot, CMBS In-Context, and CMBS Partially-Tuned. All CMBS proposals outperform the other baselines, being much cheaper.

More recently, Large Language Models (LLMs) emerged, built on top of the first generation of Transformers (aka small language models (SLMs)). Studies have pointed to LLMs as the current SOTA for several NLP tasks (Liang et al., 2023). Although the literature reports LLMs superiority for tasks such as summarization and translation, for others, such as sentiment analysis (one of our focuses), it is not yet clear whether LLMs complexity and size (e.g., in terms of number of parameters) translate into statistical and mainly *practical* gains. In fact, several studies point to the SLM RoBERTa as a very strong sentiment classifier (Cunha et al., 2023b) ranking prominently on leaderboards such as the GLUE benchmark[2].

Depending on the type of training (or its absence), LLM approaches can be divided into four groups: *zero-shot, in-context, partially- and fully-tuned*. In a zero-shot approach, the model is expected to perform tasks without specific training. In an in-context approach, the model is given a small number of examples via prompt, providing a

[1] Code available at https://github.com/Anonymous

[2] https://gluebenchmark.com/leaderboard/

context to learn from. Partially-tuned approaches use a considerable part of the training set (though not all) to generate the model (e.g. (Cunha et al., 2023b,a, 2024), while fully-tuned approaches use the complete training, allowing for better model optimization. As we shall see in our experiments, most gains of open LLMs over SLMs are obtained in the **fully-tuned** scenario at a **very high cost**.

In light of the above discussion, the first research question our paper aims to answer is **RQ1:** "Are (open) LLMs more effective in overcoming the limits of SLMs in sentiment and topic classification?". Very recent work (Fields et al., 2024) has shown no consensus as to whether LLMs always perform better in classification tasks. To address this question, we performed a comprehensive set of experiments comparing three popular SLMs (BERT, RoBERTa, and BART) and three open LLMs (LLaMA 3.1 8B, Mistral 7B, Falcon 7B) using a benchmark comprising 9 sentiment and 4 topic analysis datasets with different characteristics. Two of these datasets, in particular, were collected after the release of the LLMs (IMDB2024, RottenT2024) to minimize potential data contamination (Liang et al., 2023). In this comparison, we focus on open-source LLMs, as closed-source and proprietary LLMs, such as ChatGPT, are black boxes that prevent us from understanding how they were trained or their internal structure[3]. Our results indicate that open LLMs can outperform SLMs, reaching up to 8.3% of effectiveness gains (on average, 3.1%), though mostly in the fully-tuned mode.

Given the (much) higher computational costs associated with fully fine-tuning open LLMs (the most effective approach), a natural question we posited is **RQ2:** "How does the computational cost of using open LLMs for ATC compare to SLMs' cost?". To answer this question, we conducted a thorough analysis of our experimental results, considering zero-shot, in-context, partially-tuned, and fully-tuned strategies, to assess the trade-offs between effectiveness and costs in terms of computational time to train the models and their impact on carbon emission. We found that LLMs are orders of magnitude more costly to fully fine-tune when compared to SLMs – fully fine-tuned LLMs are up to 1700% more expensive than SLMs. As current LLMs can produce just moderate gains over SLMs and only through highly costly full fine-tuning processes, depending on the application scenario, the benefits may not be worth the costs.

All this leads to our final research question **RQ3:** "Is it possible to perform a combination of SLM and (open) LLMs to achieve a better effectiveness/cost trade-off compared to using either SLM or LLM alone?" To answer this question, we propose a novel confidence-based strategy called "Call My Big Sibling" (**CMBS**), which smoothly combines SLM and (open) LLMs **based on calibrated uncertainty**.

In CMBS, we rely on *fully fine-tuned SLMs*, which have already attained effectiveness and efficiency and are calibrated[4] for ATC tasks. We then use the classification confidence to determine whether the LLM should classify a doubtful document. In other words, the fully-tuned SLM classifies high-certainty documents (i.e., with high certainty calibrated scores), while low-certainty documents are sent to the *zero-shot, in-context or partially-tuned versions of the LLMs* for ATC. Such combination with a cheaper LLM version (compared to the fully-tuned) brings potential effectiveness gains to the SLM and is very competitive to the fully-tuned LLM, being an attractive, cost-effective option in most cases.

In more detail, our experimental results show that, for sentiment classification, the combination of a SLM with a zero-shot LLM (aka **CMBS Zero-shot**) is enough to produce gains in effectiveness at the lowest cost, highlighting the practicality of our proposal. To illustrate our argument, Figure 1 presents the effectiveness (Macro-F1) and efficiency (Time(s)) of our solution compared to the baselines in two datasets. Our proposals are highlighted with star icons in Figure 1, in Figure 1a we observe that CMBS Zero-Shot matches the effectiveness of the most computationally expensive solution, Fully-Tuned LLaMA, at a fraction of the cost. Similarly, in Figure 1b, all CMBS methods outperform the other baselines, being much cheaper.

Our experiments reveal that CBMS Zero-shot outperforms the SLM in 8 out of 9 sentiment datasets, tying in remaining one, with an increase in computational cost over SLMs of only 8%. Moreover, compared to fully-tuned LLaMA, CBMS Zero-Shot delivers comparable effectiveness at a significantly lower cost. In 4 of the 9 sentiment datasets, CMBS Zero-Shot ties with the fully-tuned

[3]Closed LLMs are irreproducible (Gao et al., 2024).

[4]The confidence of the SLM´s softmax function is highly calibrated as we shall discuss.

LLM, with minimal losses (on average, just 2%) in the other datasets, at $\frac{1}{10}$ of the cost. Moreover, CMBS Partially-Tuned ties with fully-tuned LLaMA in *all* sentiment datasets at half of the cost.

For topic classification, with a larger number of classes (up to 11 in one of our datasets) and more uneven distributions, the CBMS zero-shot version, or even the version that sends the doubtful cases to the in-context LLM (aka **CBMS In-Context**), struggles to achieve good effectiveness. Only when combined with the partially-tuned CMBS (aka **CMBS Partially-Tuned**) can we produce gains over the SLM. Among the four evaluated topic classification datasets, CMBS Partially-Tuned outperforms RoBERTa in two datasets and ties in the other two. Compared to the fully-tuned LLaMA, CMBS Partially-Tuned achieves statistical parity in three datasets (with just a 2% loss in the fourth) while operating at approximately half of the computational cost.

In sum, the main contributions of this paper are:

- We perform a comprehensive comparative evaluation of SLMs and (open) LLMs regarding cost/effectiveness trade-offs.
- We propose "Call My Big Sibling" (**CMBS**), a confidence-based strategy to combine calibrated SLMs and LLMs aimed at optimizing the effectiveness-cost trade-off.
- We perform a thorough evaluation of our three proposals considering 13 distinct datasets, in two tasks: sentiment (binary) and topic (multi-class) classification tasks, three SLMs and zero-shot, in-context, partially-tuned and fully-tuned versions of three open LLMs.

2 Related Work

LLMs' computational costs have led to numerous studies highlighting their financial and environmental impacts. For instance, (Strubell et al., 2019) illustrates the substantial financial costs propelled by the continuous need for investment in specialized hardware to manage progressively larger language models. This trend not only limits access to these models but also escalates energy consumption, affecting the environment by increasing carbon dioxide (CO_2) emissions.

Among LLMs, there are proprietary and closed-source ones, such as *chatGPT*, which operate as black boxes. This opacity poses challenges in comprehending their training methodologies or internal structures, thereby obstructing reproducibility in research reliant on these models. Moreover, utilizing such LLMs often entails transmitting data through web platforms or APIs, a delicate issue when data is sensitive and cannot be shared. As a result, numerous studies advocate for restricting scientific evaluations to run locally, open-source LLMs such as Bloom and LLaMA 2 (Spirling, 2023).

A close work to ours is (Xu et al., 2024), which also combines SLMs with LLMs for various NLP tasks, aiming to improve effectiveness. In that work, computational costs are not evaluated and the use of the LLM is not restricted to classifying a subset of hard instances; instead, it encompasses the entire test set. On the other hand, we select only low-confidence documents to be forwarded to the LLM, a strategy that greatly reduces computational costs as the LLM is significantly more expensive. Moreover, in (Xu et al., 2024), a closed LLM is employed via an API, which provides no control over the computational structure or the model architecture. Furthermore, only a single sentiment classification dataset is utilized in the experiments.

Liang et al., 2023 investigate various LLMs across multiple tasks, prompts, metrics, and datasets. Like Liang et al., we include LLM evaluation and the trade-off between efficiency and effectiveness. Unlike their study, which focuses on the breadth of evaluation with several domains (including only one sentiment dataset), our work is depth-oriented into the specific task of sentiment and topic classification, covering multiple datasets with diverse characteristics and domains. Moreover, although Liang et al. evaluates several models, they do not compare them with an SLM such as RoBERTa, considered SOTA in sentiment and topic classification (Bai et al., 2023a; Cunha et al., 2021a, 2020; França et al., 2024; Belém et al., 2024). Finally, they do not provide any solution for the effectiveness-cost trade-off problem. ***We do!***

3 The CBMS Solution

One of the main contributions of our work is the proposal of a novel strategy to combine simpler, more efficient, but perhaps less effective SLMs with potentially more effective but costly LLMs, aiming to promote effectiveness while minimizing computational costs. Our solution, "Call-My-Big-Sibling" (CMBS), metaphorically conjures up the image of a small (but smart) child who, in a challenging situation, seeks help from a bigger sibling. CBMS pursues the best trade-off between effectiveness and costs with a

confidence-based pipeline of Language Models.

CMBS seamlessly integrates SLMs and (open) LLMs by leveraging uncertainty. In this framework, we first employ *fully-tuned SLMs models*[5], which are already highly effective in some classification tasks (and faster to tune compared to LLMs). In our solution, (test) documents classified below a certain confidence threshold (a method's parameter) by the SLM are sent to an open LLM to be classified. The procedure is illustrated in Figure 2.

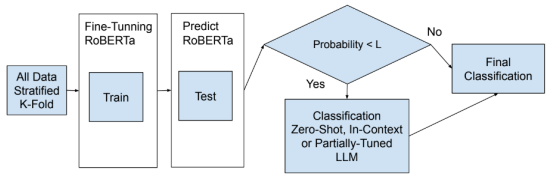


Figure 2: Flowchart of the Evaluation Methodology.

For CBMS to properly work, we have to trust the probability outputs, or, in other words, the probabilities need to be calibrated[6]. Wolfe et al., 2017 argue that RoBERTa's softmax function provides calibrated probabilities as it is a generalization of logistic regression. To demonstrate that, Table 1 presents the Brier score used to measure model calibration, in three datasets used in our experiments, by three classifiers. This score is calculated based on the model probabilities and actual labels. The score ranges from 0 to 1, with values closer to 1 indicating a better alignment between probabilistic predictions and actual outcomes. As we can observe, the table reinforces that RoBERTa is a very calibrated model (Brier score > 0.90), being as calibrated as known calibrated classifiers such as Logistic Regression and Random Forests (Cunha et al., 2024).

Dataset	RoBERTa	Random Forest	Logistic Regression
Imdb	0.941	0.940	0.938
PangMovie	0.909	0.910	0.902
Finance	0.979	0.982	0.982

Table 1: Brier score in validation set.

We select a document set for which the classifier is most uncertain about its classification (Probability < L) to send to an LLM for final prediction. Due to computational costs, we employ either the zero-shot, in-context or partially-tuned strategies for this LLM. Finally, our final prediction set is built using the following procedure: 1) we evaluate the probability the model provides and compare it with the threshold parameter; 2) we decide whether the prediction will be made using an SLM or an LLM (zero-shot, in-context or partially-tuned).

In this proposal, the choice of the confidence threshold L is essential for evaluating the documents that will be sent to the LLM. To illustrate this point, Figure 3 presents the effectiveness based on prediction confidence for the SST2 dataset. On the Y-axis, we have RoBERTa´s effectiveness, and on the X-axis, we have confidence. We can observe that the more confident, the more effective RoBERTa is in its predictions. The figure highlights the importance of selecting an appropriate confidence threshold, showing that it is more advantageous to forward low-confidence documents, as high-confident ones are classified with high accuracy by the cheaper SLM.

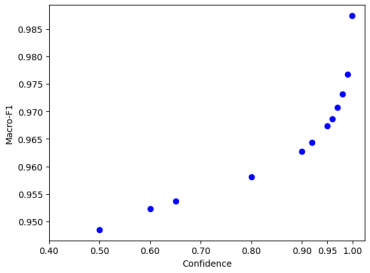


Figure 3: RoBERTa's Macro-F1 vs Confidence for SST2.

4 Experimental Methodology and Setup

4.1 Datasets

Our study draws on **thirteen** datasets developed for sentiment analysis and topic classification. The sentiment analysis datasets include **Finance**, **IMDB**, **PangMovie**, **SemEval2017**, **SST**, **SST2** and **Yelp Review** (Yelp2L), while for topic classification we used **ACM**, **DBLP**, **Twitter** and **Webkb**. With the significant amount of data used in building LLMs, several authors express concerns about contamination in evaluation data. Intending to minimize this issue, we collected and curated two datasets with data post-LLMs release (**RottenT2024** from Jan-Nov 2024 and **IMDB2024** from Jan-May 2024), ensuring no contamination in the training of these LLMs. Several works in ATC have used most of these datasets as benchmarks. See Appendix B for further information about the datasets, including domain, number of documents, density, and skewness (class imbalance). Our benchmark covers a wide variety of heterogeneous scenarios.

[5]Tuned with the full training data.

[6]A classifier is calibrated if there is a strong correlation between its class prediction probabilities and the frequency with which it correctly predicts instances belonging to each probability range.

4.2 Prompt Template

We investigated the performance of three open LLMs: FaLcon 7B, Mistral 7B, and LLaMA 3.1 8B, adopting the same prompt template utilized by (Liang et al., 2023), who, upon evaluating many alternatives, concluded that the most effective prompt contains: (i) the task description; (ii) examples with respective expected responses; and (ii) the text to be evaluated. We adapted and used such prompt for sentiment classification as illustrated in Table 6 and topic classification in Table 7 in the Appendix A. Our prompt consists of instructions and examples of classes and concludes with the text to be evaluated (Evaluate Text). Subsequently, the LLM generates the class ("next word") for the evaluated text (Response from LLM). The template for In-Context LLM (Table 8) is a small variation of the above, in which the generic example is substituted by the closest training document, according to a cosine similarity calculated using the RoBERTa's document embeddings, using an encoder generated by fine-tuning RoBERTa.

4.3 Zero-shot, In-context, Partially-tuning or Fully-tuning for Text Classification

Applying SLM or LLMs pre-trained models to ATC can be done through four strategies: zero-shot, in-context, partially-tuned, and fully-tuned. **Zero-shot** strategy predicts text classes without using training examples or performing model fine-tuning. In an in-context approach, the model relies on a prompt containing the nearest neighbors of the evaluated example inserted into the prompt to provide context for making predictions without adjusting its weights. In the **partially-tuned** strategy, a portion of labeled data is employed to adjust the model weights, simulating a scenario of data scarcity. For the partially-tuned approach, we fixed 50% of the training partition data for model training. We use this percentage inspired in recent work in Instance Selection (Cunha et al., 2023a) that determined this is an empirical threshold that assures good efficiency with minimal effectiveness losses.[7] In any case, an evaluation using different training data sizes is conducted and presented in Appendix E. Lastly, the **fully-tuned** strategy utilizes all available labeled data in the model's training partition to maximize model adjustment for the task and data domain. While this strategy typically achieves better effectiveness, it has a very high computational cost. In our paper, the fully-tuned is used to compare with our approaches

We only employ the SLMs fully tuned, which is essential for their effectiveness (de Andrade et al., 2023). Fully tuning SLMs involves fine-tuning the SLM's text representation (CLS token) and fully connected layer that predicts the text class distribution, utilizing all available training samples. Aiming to investigate the trade-off between efficiency and effectiveness within our CMBS solution, we explored the four strategies for LLMs: (i) zero-shot; (ii) in-context approach; (iii) partially-tuned using 50% of the training samples; and (iv) fully tuned.

4.4 Method-Specific Parameter Tuning

All data is divided using stratified 5-fold cross-validation, a widely accepted technique in model evaluation. This method enhances the robustness and reliability of the model by splitting the dataset into five parts: three for training, validation, and testing. In each of the five iterations, the roles of the partitions alternate between training, validation, and testing, ensuring that the class distribution is preserved in the test partition. The validation set is crucial for parameter tuning, as detailed below.

For SLMs, we adopted the hyper-parameterization suggested by (Cunha et al., 2023b), fixing the learning rate in 2×10^{-5}, the batch size with 64 documents, adjusted the model for five epochs and set the maximum size of each document to 256 tokens. For the LLM models, we adopted the following parameters: all LLMs utilize 4-bit quantization, enabling fine-tuning to be performed on reasonably equipped machines. For LLaMA, we used 1024 maximum tokens, a learning rate of 2×10^{-4}, and a temperature equal to 0.6. All other parameters were set at their default values. For fully-tuning processes, which are more costly due to the model's weight adjustment (backpropagation), we had to reduce the maximum number of tokens to 256. We performed training for three epochs.

We also introduce a confidence threshold parameter. Class predictions in which the SLM model's confidence falls below this threshold are forwarded to the LLM, which we refer to as the "Big Sibling". This term is used to illustrate the model's decision-making process, where the more complex LLM takes over when the simpler SLM is uncertain. We employ the validation set and perform classifications while varying this parameter to determine the

[7] We did not exploit Instance Selection in this work, but intend to do it in future work.

optimal threshold to maximize Macro-F1 without increasing the cost. The chosen threshold for a sample of datasets is shown in Table 13 (Appendix F), which also shows the percentage of validation instances sent to the LLM relative to the total, and the LLM and SLM effectiveness on this subset of instances. For instance, in SST2, if prediction confidence is below 0.9, the document is forwarded to the LLM; otherwise, the SLM classifies it. As the threshold increases, more documents are sent to the LLM. It is interesting to notice that the choice of L (around 0.9) that optimizes the tradeoff is similar in all datasets, and that the LLM effectiveness in these hard-to-classify instances is better than SLM's, justifying potential CBMS gains.

4.5 Metrics and Experimental Protocol

We evaluated SLMs and (open) LLMs regarding the effectiveness/cost trade-off. All models were assessed on identical hardware configuration: a 4-core processor, 32GB of system memory, and an Nvidia Tesla P100 GPU. Classification effectiveness is assessed using Macro-F1 due to imbalance in several datasets. To ensure statistical validity of the results and demonstrate model generality, models were evaluated using the test set from a 5-fold stratified cross-validation methodology and a t-test with 95% confidence with Bonferroni correction to account for multiple comparisons.

To analyze the cost-effectiveness trade-off, we also evaluated each method's cost in terms of the total time required to build the model. More specifically, the total time comprises the time for model learning (if applicable), together with the time for class prediction (considering the full test set). In the case of our CMBS solution, the CMBS Zero-Shot model building time includes the time required to fully-tuned the SLM, the time to predict a portion of the test set using the SLM, and a smaller portion using the LLM. For CMBS in-context, it also includes the time to find the k nearest neighbors. For CMBS partially-tuned, it also includes the time to train the LLM using 50% of the training data (further details provided in Appendix E).

5 Experimental Results and Analyses

5.1 RQ1: Are (open) LLMs more effective in overcoming the limits of SLMs?

To address RQ1, we first evaluated several popular open-source LLMs, including Falcon 7B, Mistral 7B, and LLaMA 3.1 8B. We began by comparing the performance of these three LLMs on sentiment and topic classification tasks in the Zero-Shot setting. Table 10 in Appendix C presents the Macro-F1 scores, highlighting the best results in bold, including statistical ties. LLaMA 3.1 8B consistently achieves the best results (statistically) across most datasets in both sentiment and topic tasks. Due to the high computational cost of fully tuning LLMs, we selected LLaMA 3.1 8B for all subsequent tests.

Regarding SLMs, we did a similar experiment and compared three widely used SLMs – BERT, BART, and RoBERTa. Results in Table 11 in Appendix D show that, among SLMs, RoBERTa achieves the highest effectiveness (or ties for it) in all cases, confirming findings reported in the literature (Cunha et al., 2023b; Bai et al., 2023b).

Finally, to explicitly answer RQ1, we compare RoBERTa with four LLaMA versions in Table 2 – zero-shot, in-context, partially-tuned with 50% of training and fully-tuned, using the entire training partition. Some interesting observations can be drawn from the Table. Zero-shot LLaMA 3.1 ties or is worse than RoBERTa in all sentiment datasets, being much worse than RoBERTa in topic classification. Similarly to Zero-Shot LLaMA, In-Context LLaMA ties or underperforms across all sentiment classification datasets compared to RoBERTa. For topic classification, the performance gap is even more pronounced. This difference can be attributed to the higher number of classes and the increased complexity of associating a document with its corresponding class in topic classification tasks.

Only partially and fully-tuned LLaMA can outperform RoBERTa, with an obvious advantage for fully tuning at double the cost. However, there are several datasets, such as Finance and Yelp2L for sentiment and Twitter for topics, which are fully-tuned and statistically tied with RoBERTa. In several other datasets, their effectiveness is also very close. This further motivates us to combine SLMs and LLMs with our proposed CMBS pipeline for the sake of optimizing the effectiveness-cost trade-off. This trade-off is the core of our subsequent analyses.

5.2 RQ2: How does the computational cost of open LLMs and SLMs compare?

Table 3 presents total time (in seconds) required to obtain final predictions for each solution. The Table shows that RoBERTa's time is the shortest, followed by LLM Zero-Shot, which is approximately 76% more expensive than the SLM, on aver-

Dataset	RoBERTa	Zero-Shot LLaMA	In-Context LLaMA	Partially-Tuned LLaMA	Fully-Tuned LLaMA
Finance	**98.1±1.9**	95.4±1.2	**98.6±1.8**	**98.6±1.2**	**98.7±1.6**
Imdb	93±0.5	93±0.3	78.9±1.2	**95.8±0.2**	**95.9±0.4**
PangMovie	88.7±0.9	88.8±0.9	89.9±0.7	93.1±0.4	**93.7±0.5**
SemEval2017	91.2±0.7	89.7±0.6	90.1±0.7	92.7±0.6	**93.5±0.3**
Sst	87.3±1	87.9±0.7	88.5±1	**90.9±0.8**	**91.1±1**
Sst2	94.6±0.2	91.4±0.4	93.3±3.9	**95.7±0.2**	**96±0.1**
Yelp2L	**97.9±0.5**	**98.5±0.3**	92.1±1	**98.5±0.6**	**98.5±0.5**
IMDB2024	97.6±1	96.5±1	93.9±1	**98.6±0.7**	**98.7±0.7**
RottenT2024	93.7±1.1	95.2±1.4	95.3±1	**96.6±0.7**	**96.7±0.4**
ACM	70.7±1.5	35.6±1.1	50.5±1.6	72.4±1.6	**76.6±2.1**
DBLP	81.9±0.7	53.7±0.8	53.2±1.0	85.9±0.8	**87.8±0.7**
Twitter	**77.5±2.7**	67.4±2.7	72.9±1.6	73.5±3.1	**77.7±2.5**
Webkb	82.3±2.6	41.9±1.5	64±1.8	82.4±2.1	**86±1.3**

Table 2: Average Macro-F1 and 95% confidence interval for SLMs and versions Llama 3.1 8B. Best results (including statistical ties) are marked in **bold**.

age. LLM In-context, in turn, is 176% slower than RoBERTa and 56% costlier than LLM Zero-Shot.

In the partially-tuned version, the cost increases significantly due to the weight adjustment process performed via backpropagation in the LLMs. Of course, the fully-tuned LLM is the most expensive solution, which is 1700% more expensive than RoBERTa. With an average improvement of 3.3% across all datasets (peaking at 8.3% in ACM), it raises the question of whether such improvements justify the significant computational cost.

Depending on the scenario in which LLMs are employed, costly solutions may not be ideal or even feasible. To address this, our proposed solution aims to reduce costs associated with using LLMs while preserving their effectiveness gains.

Dataset	RoBERTa	Zero-Shot LLaMA	In-Context LLaMA	Partially-Tuned LLaMA	Fully-Tuned LLaMA
Finance	79.1	103.4	123.3	484.2	896.4
Imdb	2615	6295.2	11548.1	25175.8	39257
PangMovie	934.4	1199.6	1489.6	5891.6	10921.1
SemEval2017	2416.4	3160.1	4250.8	15153.7	28086.7
Sst	1027.2	1229.8	1561.8	6544.2	11790.7
Sst2	5816.6	7799.8	10936.4	37434.9	26171.3
Yelp2L	510	1161.2	1736.3	2406.9	5116
IMDB2024	681.3	1622.9	2538.4	5822.2	12303.5
RottenT2024	788.7	982.8	1708.4	4393.3	8129.7
ACM	2664.7	3163.2	7896	16876.7	28206.9
DBLP	4140.1	8112.8	17311.4	27564.2	139249.5
Twitter	650.7	891.8	1477.9	6663.9	11405.9
Webkb	909.8	2877.2	3273.7	10149.9	26020.7

Table 3: Average Total Time for RoBERTa and Llama3.18B.

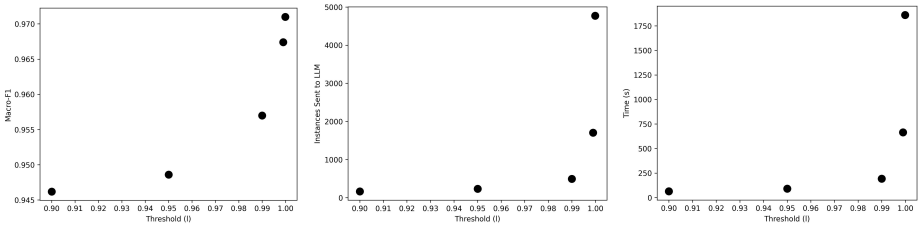
5.3 RQ3: Is it possible to perform a combination of SLM and (open) LLMs to achieve a better effectiveness/cost trade-off compared to using either SLM or LLM alone?

Let us focus now on our proposed method: **CMBS** and compare the three alternative implementations of our solution: CMBS Zero-Shot, CBMS In-Context and CBMS Partially-Tuned. Starting with the sentiment classification task, Table 4 presents the results of the SLM RoBERTa, each CMBS version, and Fully-Tuned LLaMA. We start by noticing that, CMBS Zero-Shot outperforms RoBERTa in 8 out 9 sentiment datasets, tying only in Finance. These gains come with a small increase in computational cost over SLMs of only 8%. Moreover, in 4 of the 9 datasets, CMBS Zero-Shot ties with the fully-tuned LLM, with minimal losses in others (on average, just 2% less effective). These excellent effectiveness results come at $\frac{1}{10}$ of the cost, as demonstrated in Table 5, which presents total time results for all alternatives. Moreover, CMBS partially-tuned ties with fully-tuned LLama in *all* sentiment datasets at half of the cost.

For topic classification, characterized by datasets with a larger number of categories (up to 11 in one case) and highly uneven class distributions, CBMS Zero-Shot, as well as **CBMS In-Context**, encounter significant challenges in maintaining high levels of effectiveness. Significant improvements over the SLM are obtained only when leveraging the partially-tuned variant (**CBMS Partially-Tuned**). Among the four topic classification datasets analyzed, CMBS Partially-Tuned model surpasses RoBERTa in effectiveness on two datasets and ties it on the remaining two. Indeed, CMBS Partially-Tuned can improve over partially-tuned LLM is all topic datasets, with gains of up to 6.4% in Twitter. Finally, when compared to the fully-tuned LLaMA model, CMBS Partially-Tuned demonstrates statistical equivalence across three datasets and incurs only a marginal 2% effectiveness deficit on the fourth, all while reducing computational demands by approximately 50%.

Summarizing, for sentiment analysis, the best effectiveness tradeoff is achieved by CBMS Zero-Shot. If effectiveness is mandatory, the choice is CBMS Partially-Tuned, which has the same effectiveness of LLaMA Fine-tuned at half of the cost. For topics, the choice is also CBMS Partially-Tuned, which ties with LLaMA fine-tuned in 3 out four datasets, loosing minimally (by 2%) in the 4th dataset, being twice more efficient.

We also calculated the CO_2 emissions associated with obtaining final model predictions, using the methodology developed by Lacoste et al. (2019). Table 14 (Appendix G) presents the results. Similar to time, CO_2 emissions are much higher for LLMs, in this case, by orders of magnitude. Financial costs associated with the solutions are also analyzed in Appendix H, with similar conclusions.



(a) Effectiveness (b) Instances Sent to LLM (c) Efficiency

Figure 4: Effectiveness, Size of the Test Set Sent to LLM and Efficiency for IMDB dataset.

Dataset	RoBERTa	CMBS Zero-Shot	CMBS In-Context	CMBS Partially-Tuned	Fully-Tuned LLaMA
Finance	**98.1±1.9**	**98±2.1**	**98.2±1.7**	**98.3±1.3**	**98.7±1.6**
Imdb	93±0.5	94±0.6	92.5±0.6	**95.8±0.2**	**95.9±0.4**
PangMovie	88.7±0.9	90.2±0.9	89.9±0.8	**93.1±0.3**	**93.7±0.5**
SemEval2017	91.2±0.7	92.2±0.6	92±0.5	**92.9±0.6**	**93.5±0.3**
Sst	87.3±1	89±0.6	88.5±1.2	**90.9±0.9**	**91.1±1**
Sst2	94.6±0.2	95.1±0.2	94.8±0.3	**95.7±0.2**	**96±0.1**
Yelp2L	97.9±0.5	**98.5±0.2**	**98.1±0.2**	**98.6±0.5**	**98.5±0.5**
IMDB2024	97.6±1	**98.2±0.9**	97.3±1.2	**98.7±0.8**	**98.7±0.7**
RottenT2024	93.7±1.1	95.6±1	95.7±0.7	**96.3±0.7**	**96.7±0.4**
ACM	70.7±1.5	70.5±1.2	70.6±1.2	**73.3±2.4**	**76.6±2.1**
DBLP	81.9±0.7	81.9±0.6	82±1.6	86±0.8	**87.8±0.7**
Twitter	**77.5±2.7**	**79.4±2.7**	**78.7±2.5**	**78.2±1.8**	**77.7±2.5**
Webkb	82.3±2.6	82.1±2.3	82.2±2.7	**83.8±2.5**	**86±1.3**

Table 4: Average Macro-F1 and 95% confidence interval for RoBERTa, Zero-Shot LLaMA 3.1 and CMBS Zero-Shot. Best results (including statistical ties) are marked in **bold**.

Dataset	RoBERTa	CMBS Zero-Shot	CMBS In-Context	CMBS Partially-Tuned	Fully-Tuned LLM
Finance	79.1	84.27	89.44	514.88	896.4
Imdb	2615	2929.76	3244.52	25273.22	39257
PangMovie	934.4	994.38	1054.36	6236.84	10921.1
SemEval2017	2416.4	2574.405	2732.41	16054.73	28086.7
Sst	1027.2	1088.69	1150.18	6916.98	11790.7
Sst2	5816.6	6206.59	6596.58	39508.01	26171.3
Yelp2L	510	568.06	626.12	2676.21	5116
IMDB2024	681.3	762.445	843.59	5921.28	12303.5
RottenT2024	788.7	837.84	886.98	4742.67	8129.7
ACM	2664.7	2822.86	2981.02	17853.73	28206.9
DBLP	4140.1	4545.74	4951.38	28947.88	139249.5
Twitter	650.7	695.29	739.88	6648.21	11405.9
Webkb	909.8	1053.66	1197.52	10044.71	26020.7

Table 5: Average Total Time total- RoBERTa, CMBS(Zero-Shot,In-Context, Partially-Tuned) and Fully-Tuned LLM.

5.4 Confidence Threshold Sensitivity Analysis

We analyze the role of the uncertainty threshold in the results. For this, we show in Figure 4, the results in the IMDB, a dataset in which CMBS (Partially-tuned) obtained one of the best cost-benefit tradeoffs: it improves effectiveness over RoBERTa and ties with Fully-tuned LLaMA, while requiring only half the computational cost.

Figure 4a, 4b and 4c show effectiveness increase, number of instances sent to LLM, and respective cost increase. It is interesting to see that the patterns in increase are very similar in the three graphs, although the metrics are very different. We can also see that by choosing appropriate thresholds, there is still room for effectiveness improvements, although at the expense of cost increases.

6 Conclusion

We proposed Call-My-Big-Sibling (CMBS), a novel ATC solution combining already very effective, efficient, and calibrated SLMs with more effective but costlier open LLMs, aiming at optimizing an effectiveness-cost trade-off. Our approach involves leveraging LLMs only when the SLM exhibits high uncertainty in its calibrated predictions.

Experiments conducted on 13 diverse datasets on sentiment and topic classification underscored the superiority of our solutions regarding the aforementioned trade-off. For sentiments, CBMS Zero-Shot outperformed the SLM in 8 out 9 datasets, tying in the other, with a marginal increase in computational time, while also being very competitive with fully-tuned LLM. CMBS Partially-Tuned, in turn, matches the fully-tuned LLM in all sentiment datasets, improving over partially-tuned LLaMA, at half of the cost of the fully-tuned LLM.

Similar results are obtained for topics, in which CMBS Partially-Tuned improves over partially-tuned LLaMA in all datasets (up to 6.4%) and is as good as the fully-tuned LLM in 3 out 4, almost tying (less than 2% of effectiveness loss) in the fourth dataset, while being twice more efficient. In real-world, practical scenarios, such minimal effectiveness difference may not impact any application or user, while a cost difference of 50% may bring many practical benefits. These results confirm our hypothesis that CBMS, which leverages a confidence-based combination of SLMs and LLMs, can achieve a better effectiveness-cost balance than the two isolated components of the solution.

In future work, we will apply CMBS to other ATC tasks, such as hate speech and irony detection, as well as to other NLP tasks, such as summarization and Q&A. We intend to apply instance selection (Cunha et al., 2023b,a, 2024) to further reduce the amount of training in CBMS Partially-tuned, improving even further its efficiency, while keeping similar effectiveness levels. Finally, we intend to run tests with other LLMs and configurations.

7 Limitations

Despite relevant contributions, our study has some limitations. Our current work covers only two classification tasks, which we have pursued to evaluate in depth. In this study, we used 13 datasets, 9 on sentiment analysis and 4 on topic classification, all with distinct characteristics.

We focused our evaluation on open LLMs for the sake of the reproducibility of subsequent research using our method. Among LLMs, there are proprietary and closed-source ones, such as ChatGPT, which operate as black boxes. This opacity poses challenges in understanding their training methodologies or internal structures, thereby obstructing reproducibility in research reliant on these models.

LLMs have been made available for different purposes. Some of these LLMs have high execution costs, such as Falcon 180B (Penedo et al., 2023), which requires an expensive infrastructure to use it. In our work, we limited our study to the best evaluated LLMs in the Hugging Face system[8], with around 7 billion parameters, which have a reasonable structure allowing us to evaluate zero-shot, in-context, partially and fully-tuned versions of our solutions.

Finally, our work focused on applying our proposed solution with three open LLMs – Falcon 7B, Mistral 7B and LLaMA 3.1 8B. However, new LLMs, such as Orca and LLaMA 3.3, emerged during the development of this work, and we were not able to use them in time for reporting the results in this paper. We intend to use Orca as well as other new open LLMs that will come out in the near future. Nevertheless, considering that these new LLMs tend to be increasingly complex and costly, optimizing the cost-benefit of our combined solution between SLMs and LLMs will certainly be still valid and even a more appealing goal.

References

Dimosthenis Antypas, Asahi Ushio, Jose Camacho-Collados, Leonardo Neves, Vitor Silva, and Francesco Barbieri. 2022. Twitter Topic Classification. In *Proceedings of the 29th International Conference on Computational Linguistics*, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Jun Bai, Xiaofeng Zhang, Chen Li, Hanhua Hong, Xi Xu, Chenghua Lin, and Wenge Rong. 2023a. How to determine the most powerful pre-trained language model without brute force fine-tuning? an empirical survey. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5369–5382, Singapore. Association for Computational Linguistics.

Jun Bai, Xiaofeng Zhang, Chen Li, Hanhua Hong, Xi Xu, Chenghua Lin, and Wenge Rong. 2023b. How to determine the most powerful pre-trained language model without brute force fine-tuning? an empirical survey. In *Findings of the EMNLP 2023*.

Fabiano Belém, Washington Cunha, Celso França, Claudio Andrade, Leonardo Rocha, and Marcos André Gonçalves. 2024. A novel two-step fine-tuning pipeline for cold-start active learning in text classification tasks. *arXiv preprint arXiv:2407.17284*.

Sérgio D. Canuto, Marcos André Gonçalves, and Fabrício Benevenuto. 2016. Exploiting new sentiment-based meta-level features for effective sentiment analysis. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining, San Francisco, CA, USA, February 22-25, 2016*, pages 53–62. ACM.

Mark Craven, Dan DiPasquo, Dayne Freitag, Andrew McCallum, Tom Mitchell, Kamal Nigam, and Seán Slattery. 1998. Learning to extract symbolic knowledge from the world wide web. In *Proceedings of the Fifteenth National/Tenth Conference on Artificial Intelligence/Innovative Applications of Artificial Intelligence*, AAAI '98/IAAI '98, page 509–516.

Washington Cunha, Sérgio Canuto, Felipe Viegas, Thiago Salles, Christian Gomes, Vitor Mangaravite, Elaine Resende, Thierson Rosa, Marcos André Gonçalves, and Leonardo Rocha. 2020. Extended pre-processing pipeline for text classification: On the role of meta-feature representations, sparsification and selective sampling. *Information Processing & Management*, 57(4):102263.

Washington Cunha, Celso França, Guilherme Fonseca, Leonardo Rocha, and Marcos André Gonçalves. 2023a. An effective, efficient, and scalable confidence-based instance selection framework for transformer-based text classification. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 665–674.

Washington Cunha, Vítor Mangaravite, Christian Gomes, Sérgio Canuto, Elaine Resende, Cecilia Nascimento, Felipe Viegas, Celso França, Wellington Santos Martins, Jussara M Almeida, et al. 2021a. On the cost-effectiveness of neural and non-neural approaches and representations for text classification: A comprehensive comparative study. *Information Processing & Management*, 58(3):102481.

Washington Cunha, Vitor Mangaravite, Christian Gomes, Sérgio Canuto, Elaine Resende, Cecilia Nascimento, Felipe Viegas, Celso França, Wellington Santos Martins, Jussara Almeida, Thierson Rosa,

[8] https://huggingface.co/models?pipeline_tag=text-generation&sort=likes

Leonardo Rocha, and Marcos A. Goncalves. 2021b. On the cost-effectiveness of neural and non-neural approaches and representations for text classification: A comprehensive comparative study. *Information Processing Management*, 58(3):102481.

Washington Cunha, Alejandro Moreo, Andrea Esuli, Fabrizio Sebastiani, Leonardo Rocha, and Marcos André Gonçalves. 2024. A noise-oriented and redundancy-aware instance selection framework. *ACM Trans. Inf. Syst.* Just Accepted.

Washington Cunha, Felipe Viegas, Celso França, Thierson Rosa, Leonardo Rocha, and Marcos André Gonçalves. 2023b. A comparative survey of instance selection methods applied to non-neural and transformer-based text classification. *ACM CSUR*.

Claudio M.V. de Andrade, Fabiano M. Belém, Washington Cunha, Celso França, Felipe Viegas, Leonardo Rocha, and Marcos André Gonçalves. 2023. On the class separability of contextual embeddings representations – or "the classifier does not matter when the (text) representation is so good!". *Information Processing & Management*, 60(4):103336.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. pages 4171–4186.

John Fields, Kevin Chovanec, and Praveen Madiraju. 2024. A survey of text classification with transformers: How wide? how large? how long? how accurate? how expensive? how safe? *IEEE Access*, 12:6518–6531.

Celso França, Rennan C Lima, Claudio Andrade, Washington Cunha, Pedro OS Vaz de Melo, Berthier Ribeiro-Neto, Leonardo Rocha, Rodrygo LT Santos, Adriana Silvina Pagano, and Marcos André Gonçalves. 2024. On representation learning-based methods for effective, efficient, and scalable code retrieval. *Neurocomputing*, 600:128172.

Mingqi Gao, Xinyu Hu, Jie Ruan, Xiao Pu, and Xiaojun Wan. 2024. Llm-based nlg evaluation: Current status and challenges. *arXiv preprint arXiv:2402.01383*.

Tyler Griggs, Xiaoxuan Liu, Jiaxiang Yu, Doyoung Kim, Wei-Lin Chiang, Alvin Cheung, and Ion Stoica. 2024. M\'elange: Cost efficient large language model serving by exploiting GPU heterogeneity. *CoRR*, abs/2404.14527.

Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. 2019. Quantifying the carbon emissions of machine learning. *arXiv preprint arXiv:1910.09700*.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan, Ce Zhang, Christian Alexander Cosgrove, Christopher D Manning, Christopher Re, Diana Acosta-Navas, Drew Arad Hudson, Eric Zelikman, Esin Durmus, Faisal Ladhak, Frieda Rong, Hongyu Ren, Huaxiu Yao, Jue WANG, Keshav Santhanam, Laurel Orr, Lucia Zheng, Mert Yuksekgonul, Mirac Suzgun, Nathan Kim, Neel Guha, Niladri S. Chatterji, Omar Khattab, Peter Henderson, Qian Huang, Ryan Andrew Chi, Sang Michael Xie, Shibani Santurkar, Surya Ganguli, Tatsunori Hashimoto, Thomas Icard, Tianyi Zhang, Vishrav Chaudhary, William Wang, Xuechen Li, Yifan Mai, Yuhui Zhang, and Yuta Koreeda. 2023. Holistic evaluation of language models. *Transactions on Machine Learning Research*. Featured Certification, Expert Certification.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA*, pages 142–150. The Association for Computer Linguistics.

Pekka Malo, Ankur Sinha, Pekka J. Korhonen, Jyrki Wallenius, and Pyry Takala. 2014. Good debt or bad debt: Detecting semantic orientations in economic texts. *J. Assoc. Inf. Sci. Technol.*, 65(4):782–796.

Luiz Felipe Mendes, Marcos Gonçalves, Washington Cunha, Leonardo Rocha, Thierson Couto-Rosa, and Wellington Martins. 2020. "keep it simple, lazy" – metalazy: A new metastrategy for lazy text classification. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, CIKM '20, page 1125–1134, New York, NY, USA. Association for Computing Machinery.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 115–124, Ann Arbor, Michigan. Association for Computational Linguistics.

Andrea Pasin, Washington Cunha, Marcos André Gonçalves, and Nicola Ferro. 2024. A quantum annealing instance selection approach for efficient and effective transformer fine-tuning. In *Proceedings of the 2024 ACM SIGIR International Conference on Theory of Information Retrieval*, pages 205–214.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The refinedweb dataset for falcon LLM: outperforming curated corpora with web data, and web data only. *CoRR*, abs/2306.01116.

Sara Rosenthal, Noura Farra, and Preslav Nakov. 2019. Semeval-2017 task 4: Sentiment analysis in twitter. *CoRR*, abs/1912.00741.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1631–1642. ACL.

Arthur Spirling. 2023. Why open-source generative ai models are an ethical way forward for science. *Nature*, 616(7957):413–413.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in nlp. *arXiv preprint arXiv:1906.02243*.

Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. 2008. Arnetminer: Extraction and mining of academic social networks. KDD '08, page 990–998, New York, NY, USA. Association for Computing Machinery.

Felipe Viegas, Sergio Canuto, Washington Cunha, Celso França, Claudio Valiense, Leonardo Rocha, and Marcos André Gonçalves. 2023. Clusent – combining semantic expansion and de-noising for dataset-oriented sentiment analysis of short texts. In *Proceedings of the 29th Brazilian Symposium on Multimedia and the Web*, WebMedia '23, page 110–118, New York, NY, USA. Association for Computing Machinery.

J. Wolfe, X. Jin, T. Bahr, and N. Holzer. 2017. Application of softmax regression and its validation for spectral-based land cover mapping. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-1/W1:455–459.

Canwen Xu, Yichong Xu, Shuohang Wang, Yang Liu, Chenguang Zhu, and Julian McAuley. 2024. Small models are valuable plug-ins for large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 283–294, Bangkok, Thailand. Association for Computational Linguistics.

Bruna Stella Zanotto, Ana Paula Beck da Silva Etges, Avner Dal Bosco, Eduardo Gabriel Cortes, Renata Ruschel, Ana Claudia De Souza, Claudio MV Andrade, Felipe Viegas, Sergio Canuto, Washington Luiz, et al. 2021. Stroke outcome measurements from electronic medical records: cross-sectional study on the effectiveness of neural and nonneural classifiers. *JMIR Medical Informatics*, 9(11):e29120.

A Prompt Templates

Below are examples of the structure of the prompts we used for the Zero-Shot and In-Context versions of the LLMs in our experiments. They are all inspired by the work of (Liang et al., 2023). Table 6 provides the prompt used for sentiment classification, while Table 7 presents the prompt for topic classification. Both prompts include the tag [Evaluate Text], which represents the (test) text to be classified, and the tag [Response from LLM], which contains the model's output. If the model's output does not match any of the given alternatives (due to hallucination), we predict the majority class from the training set.

The third example of prompt, shown in Table 8, is tailored for in-context learning. For the evaluated test document, "I spent a day at a 5-star hotel, which was amazing." the most similar example from the training set included in the prompt was "5-star hotels have many food options.". For each evaluated example, a vector representation is generated using the fully-tuned RoBERTa as an encoder. By comparing the vector of the evaluated (test) document with the vectors of the training set documents, we identify the most similar document based on the cosine similarity between the vectors and use it as a training example in the prompt.

Classify the sentiment in the text exclusively as positive or negative: Input: I love you. Reference: A. Positive B. Negative Answer: A Input: The product is bad. Reference: A. Positive B. Negative Answer: B Input: {**Evaluate Text**} Reference: A. Positive B. Negative Answer: {**Response from LLM**}

Table 6: Prompt template for sentiment classification.

B Datasets

Our study draws on **thirteen** datasets developed for topic and sentiment classification. Our choice was strategically purposeful due to the effort to perform an in-depth analysis of this task. The datasets include **Finance** (Malo et al., 2014) focusing on

Classify the topic of the text exclusively with one of the references:
Input: Messi scored a goal against France.
Reference:
A. Pop culture
B. Sports or gaming
C. Daily life
D. Science or technology
E. Business or entrepreneurs
F. Arts or culture
Answer: B
Input: {**Evaluate Text**}
Reference:
A. Pop culture
B. Sports or gaming
C. Daily life
D. Science or technology
E. Business or entrepreneurs
F. Arts or culture
Answer: {**Response from LLM**}

Table 7: Prompt template for topic classification.

Classify the sentiment text exclusively as positive or negative:
Input: 5-star hotels have many food options.
Reference:
A. Positive
B. Negative
Answer: A
Input: I spent a day at a 5-star hotel, which was amazing.
Reference:
A. Positive
B. Negative
Answer: {**Response from LLM**}

Table 8: Prompt template for sentiment classification for In-Context Llama and CMBS In-Context.

economic news, **IMDB** (Maas et al., 2011)[9] compiling movie reviews as well as **PangMovie** (Pang and Lee, 2005) including Rotten Tomatoes[10] data, **SemEval2017** (Rosenthal et al., 2019) containing Twitter texts used in a significant text classification challenge, and the Stanford Sentiment Treebank (**SST**) (Socher et al., 2013) and **SST2** (Socher et al., 2013), where sentiment classification relies on *Treebank*, a corpus with sentiment labels and labeled parse trees. **Yelp Review** is a subset of Yelp data widely used in sentiment classification studies (Canuto et al., 2016; Viegas et al., 2023; Mendes et al., 2020). **IMDB2024** and **RottenT2024** were collected to avoid data contamination by LLM. For topic classification, we have ACM Digital Library (Cunha et al., 2021b), DBLP (Tang et al., 2008), Twitter Topic (Antypas et al., 2022) and WebKB (Craven et al., 1998).

[9] https://www.imdb.com/
[10] https://www.rottentomatoes.com/

	Dataset	Domain	|D|	Avg Words	Classes	Minor Class	Major Class
Sentiment	Finance	Finance	873	24.88	2	303	570
	IMDB	Movie	24904	234	2	12432	12472
	PangMovie	Movie	10662	21.02	2	5331	5331
	SemEval2017	Twitter	27413	19.85	2	7745	19668
	Sst	Movie	11841	19.18	2	5905	5936
	Sst2	Movie	66973	10.45	2	29643	37330
	Yelp2L	Place	4995	131.8	2	2495	2500
	IMDB2024	Movie	6572	163.02	2	2057	4515
	RottenT2024	Movie	7948	46.13	2	3315	4633
Topic	Acm	Article	24897	63.52	11	63	6562
	Dblp	Article	38128	141.43	10	1414	9746
	Twitter	Twitter	6997	28.68	6	152	2738
	Webkb	Pages	8199	208.81	7	137	3705

Table 9: Datasets Statistics.

As detailed in Table 9, we can observe an ample diversity in many aspects in these datasets: domain, number of documents (|D|), density (the average number of words per document), etc.

C Evaluating LLMs

We evaluate three SLMs: Falcon 7B, Mistral 7, and LLama 3.1-8B. Table 10 presents the results in terms of Macro-F1, with the best outcomes highlighted in bold. As observed, LLaMA is consistently the best performer, either alone or tied with Mistral, across **all** datasets, but WebKB.

Dataset	Falcon 7B	Mistral 7B	Llama 3.1 8B
Finance	46.7±4.8	**94.3±1.9**	**95.4±1.2**
Imdb	68.4±0.7	68.4±0.7	**93±0.3**
PangMovie	43.6±0.5	82.3±0.9	**88.8±0.9**
SemEval2017	54.4±0.6	81±0.9	**89.7±0.6**
Sst	47±1.2	82±0.8	**87.9±0.7**
Sst2	38.6±0.1	86.2±0.5	**91.4±0.4**
Yelp2L	79.9±1.3	96.2±0.9	**98.6±0.3**
IMDB2024	78.4±0.8	94.9±0.9	**96.5±1**
RottenT2024	65.8±1.3	93.8±1.2	**95.3**±1
ACM	2.6±0.2	18.2±0.9	**35.6±1.1**
DBLP	3.1±0.2	50.2±0.6	**53.7±0.8**
Twitter	13±0.3	**62.2±2.1**	**63.5±1.7**
Webkb	3.8±0.3	**42.1±0.6**	37±2.1

Table 10: Effectiveness in Macro-F1 for sentiment and topic classification tasks with the LLMs in Zero-shot version, Falcon 7B, Mistral 7B, and Llama 3.1 8B.

D Evaluating SLMs

We evaluate three SLMs: BART, BERT, and RoBERTa. Table 11 presents the results in terms of Macro-F1, with the best outcomes highlighted in bold. As observed, RoBERTa is consistently the best performer, alone or tied with another SLM, across **all** datasets, with no exception.

E Evaluating the training of the LLM

As mentioned, fine-tuning is essential for LLM effectiveness. Here, we illustrate the impact of training data size on the LLM effectiveness, using the validation set and a sample of two datasets. The

Dataset	BERT	BART	RoBERTa
Finance	94.1±3.8	**97±1.7**	**98.1±1.9**
Imdb	91.7±0.4	**92.8±0.4**	**93±0.5**
PangMovie	87.5±0.7	**88.4±1**	**88.7±0.9**
SemEval2017	90.3±0.3	**91±0.4**	**91.2±0.7**
Sst	86.1±0.4	**87.7±1.1**	**87.3±1**
Sst2	**94.8±0.1**	94.2±0.3	**94.6±0.2**
Yelp2L	96.8±0.4	**97.7±0.2**	**97.9±0.5**
IMDB2024	96.6±0.5	**97.5±0.6**	**97.6±1**
RottenT2024	92.5±1	**93.5±0.5**	**93.7±1.1**
ACM	**69.8±1.8**	68±2.8	**70.7±1.5**
DBLP	**82.1±0.9**	**81.9±0.6**	**81.9±0.7**
Twitter	**76.6±4.4**	**76.9±3.3**	**77.5±2.7**
Webkb	**80.8±3.8**	**81.7±3.5**	**82.3±2.6**

Table 11: Average Macro-F1 and 95% confidence interval for SLMs . Best results (including statistical ties) marked in **bold**.

pattern of results is basically the same in all other datasets we experimented with.

Table 12 presents the effectiveness results when utilizing 30%, 50%, and 70% of the training data in Twitter and WebKB, two topic datasets in which CMBS performs very well. As we can see in the Table, 30% of training generally is not enough for achieving reasonable effectiveness, while the improvements of using 70% are either marginal or incur in higher costs.

As discussed in Section 5, the CMBS Partially-Tuned version we employed in our experiments uses 50% of the training data based on results of instance selection experiments (Cunha et al., 2023a). In all datasets, such a choice produced the best tradeoff between effectiveness and computational cost. We should stress that the selection fo training instances is random. In future work, we will employ instance selection (Cunha et al., 2024) to evaluate whether we can reduce the training set size even further without incurring in effectiveness losses, by smartly chosen the instances to train.

Dataset	Portion Train	Macro-F1
Twitter	30	66.2
Twitter	50	71.9
Twitter	70	76.1
Webkb	30	76.7
Webkb	50	83.4
Webkb	70	85.2

Table 12: Evaluate amount training LLM.

F Evaluating Threshold L

We evaluate the impact of the parameter L, which determines the number of documents sent to the LLM. The higher the value of L, the more documents fall below the threshold, leading to an increase in the number of documents forwarded to the LLM. Table 13 presents this evaluation for a sample of four datasets, showing the dataset name, the percentage of instances sent to the LLM relative to the total of test instances, and the effectiveness of both the SLM and LLM on this subset of instances. It is interesting to notice that the choice of L that maximizes the cost-effectiveness threshold (around 0.9) is similar in all datasets, and that the LLM effectiveness in these hard-to-classify instances is better than the SLM, which justifies the CBMS gains.

Dataset	Percentage of Instances	SLM Macro-F1	LLM Macro-F1	Threshould (L)
IMDB2024	7.8	0.72	0.87	0.9
SST2	25.3	0.82	0.85	0.9
Webkb	13.9	0.56	0.67	0.9
Twitter	13.9	0.51	0.53	0.9

Table 13: Evaluate Threshould L.

G CO_2 emissions

We calculated the CO_2 emissions associated with the execution of the model using the methodology developed by Lacoste et al. (2019). It is possible to associate the value of emission 0.14 kg of CO_2eq per hour with a machine of similar structure to the one used in our experiments[11]. The emission values are presented in Table 14.

H Finance Cost

In the literature, some studies also analyze the financial costs of executing machine learning methods on cloud services (Griggs et al., 2024). Table 15 presents the financial cost in dollars for executing the main methods discussed in this paper. We used as a reference the hourly price of a setup similar to the one used in this research [12], offered by a large cloud company, which currently costs $0.752 per hour. The total cost for the main experiments amounted to $901.

[11] https://mlco2.github.io/impact/#co2eq

[12] https://aws.amazon.com/ec2/instance-types/g4/

Dataset	RoBERTa	Zero-Shot LLaMA	In-Context LLaMA	Partially-Tuned LLaMA	CMBS Zero-Shot	CMBS In-Context	CMBS Partially-Tuned	Fully-Tuned LLaMA
Finance	0.02	0.02	0.02	0.09	0.02	0.02	0.1	0.17
Imdb	0.51	1.22	2.25	4.9	0.57	0.63	4.91	7.63
PangMovie	0.18	0.23	0.29	1.15	0.19	0.21	1.21	2.12
SemEval2017	0.47	0.61	0.83	2.95	0.5	0.53	3.12	5.46
Sst	0.2	0.24	0.3	1.27	0.21	0.22	1.34	2.29
Sst2	1.13	1.52	2.13	7.28	1.21	1.28	7.68	5.09
Yelp2L	0.1	0.23	0.34	0.47	0.11	0.12	0.52	0.99
IMDB2024	0.13	0.32	0.49	1.13	0.15	0.16	1.15	2.39
RottenT2024	0.15	0.19	0.33	0.85	0.16	0.17	0.92	1.58
ACM	0.52	0.62	1.54	3.28	0.55	0.58	3.47	5.48
DBLP	0.81	1.58	3.37	5.36	0.88	0.96	5.63	27.08
Twitter	0.13	0.17	0.29	1.3	0.14	0.14	1.29	2.22
Webkb	0.18	0.56	0.64	1.97	0.2	0.23	1.95	5.06

Table 14: Emission CO_2. Calculation based on the work of Lacoste et al. (2019).

Dataset	RoBERTa	Zero-Shot LLaMA	In-Context LLaMA	Partially-Tuned LLaMA	CMBS Zero-Shot	CMBS In-Context	CMBS Partially-Tuned	Fully-Tuned LLaMA
Finance	0.08	0.11	0.13	0.51	0.09	0.09	0.54	0.94
Imdb	2.73	6.57	12.06	26.29	3.06	3.39	26.4	41
PangMovie	0.98	1.25	1.56	6.15	1.04	1.1	6.51	11.41
SemEval2017	2.52	3.3	4.44	15.83	2.69	2.85	16.77	29.33
Sst	1.07	1.28	1.63	6.84	1.14	1.2	7.22	12.31
Sst2	6.08	8.15	11.42	39.1	6.48	6.89	41.26	27.33
Yelp2L	0.53	1.21	1.81	2.51	0.59	0.65	2.8	5.34
IMDB2024	0.71	1.7	2.65	6.08	0.8	0.88	6.18	12.85
RottenT2024	0.82	1.03	1.78	4.59	0.88	0.93	4.95	8.49
ACM	2.78	3.3	8.25	17.63	2.95	3.11	18.65	29.46
DBLP	4.32	8.47	18.08	28.79	4.75	5.17	30.23	145.44
Twitter	0.68	0.93	1.54	6.96	0.73	0.77	6.94	11.91
Webkb	0.95	3.01	3.42	10.6	1.1	1.25	10.49	27.18

Table 15: Finance Cost in dollars ($) for RoBERTa, Zero-Shot LLaMA, In-Context LLaMA, Partially-Tuned LLaMA, CMBS Zero-Shot, CMBS In-Context, CMBS Partially-Tuned and Fully-Tuned LLaMA.