Joint multi-task learning and knowledge distillation for sentiment analysis and hate speech detection (final results)

Radu Breazu, Mihnea Vîrlan

28 January 2025

1 Introduction

As stated in the document of the previous stage, in this project, we are implementing a BERT-based two-head architecture for sentiment analysis and hate speech detection, the purpose being that of seeing whether the hate speech detection task benefits from using a model that is pre-trained for solving the more complex task of multi-label sentiment analysis. Furthermore, we are investigating whether the knowledge distillation paradigm is capable of producing a smaller model that is able to achieve roughly the same results on hate speech detection as a base model (that, during our testing, achieves the best performance).

The focus of this stage is the analysis of distillation methods, in order to try and find a model of a reasonable size that is capable of predicting both the degree of positivity, or of negativity, in a text and, based on this, whether a text represents hate speech or not.

2 Related Work

2.1 Multi-task learning

Multi-task learning comes into different flavours. This typically means having a set of shared parameters. The number of shared parameters can be dynamically adjusted [13]. The usage of certain layers is determined by a score, and once a threshold is crossed, the layer becomes task specific. Shared layers can help in incorporating common features and they can be combined with specific features for each task, an approach used in [14] which uses Bidirectional LSTM's in a shared-private configuration. An alternative to BiLSTM's is BERT [6], and can be used not only when multi-tasking similar tasks, but also heterogenous combinations such as text classification and named entity recognition, like in [2], where the embedding of the [CLS] is used to predict the news category, and the rest of the embedding for named entity recognition. What's also different there is that they scale the gradients for the tasks. Another approach highlighted in [11] consists of treating all the tasks as part of a single one, namely question answering, by considering a specific question for each task, which is prepended to the context.

2.2 Knowledge distillation

Knowledge distillation is a paradigm in which a bigger model, named the teacher, compresses the knowledge learned such that a smaller model, named the student gets to learn from it. At first, it can be said that the difference in probability distributions for the models must be optimized [8]. In the work of Tang et al. [12], a BiLSTM was used as a student network, learning from a BERT teacher network and the loss used incorporated the L2-norm of the output logits of the models. When it comes to loss calculation, [9] considers that the contributions of the loss between the models and the task-specific loss can be learned during training alongside the specific model training. A distillation can involve multiple tasks which contain similarities and an approach highlighted in [1] consists of projecting the output of the teacher model results into the student's model space and using the Singular Value Decomposition in the loss function.

3 Methodology

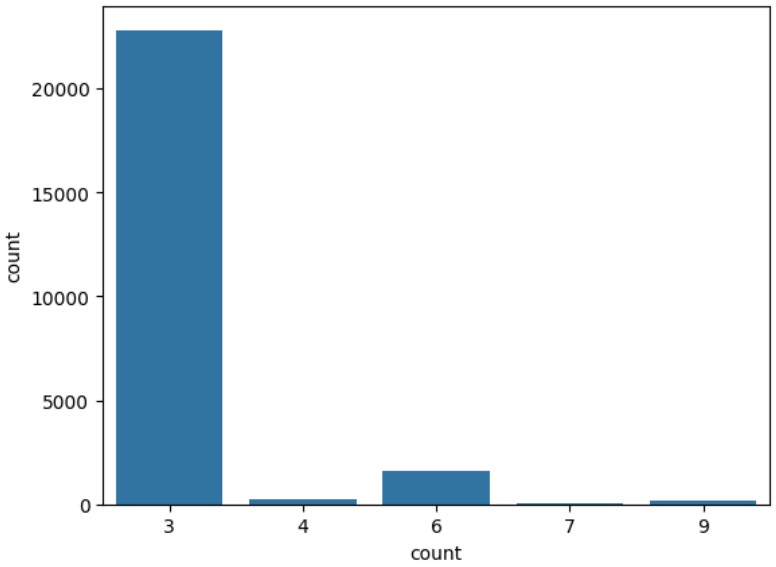
3.1 Datasets

For the hate speech detection task, we have replaced the dataset we used in the last stage with the true Davidson dataset [5], which is the one found at [7]. This dataset contains approximately 24 000 tweets, each of which has been annotated by at least 3 coders and categorized as hate speech, offensive speech or neither. Approximately 5% of the texts in the dataset fall into the hate speech category. For the sentiment analysis task, we have kept the IMDb dataset.

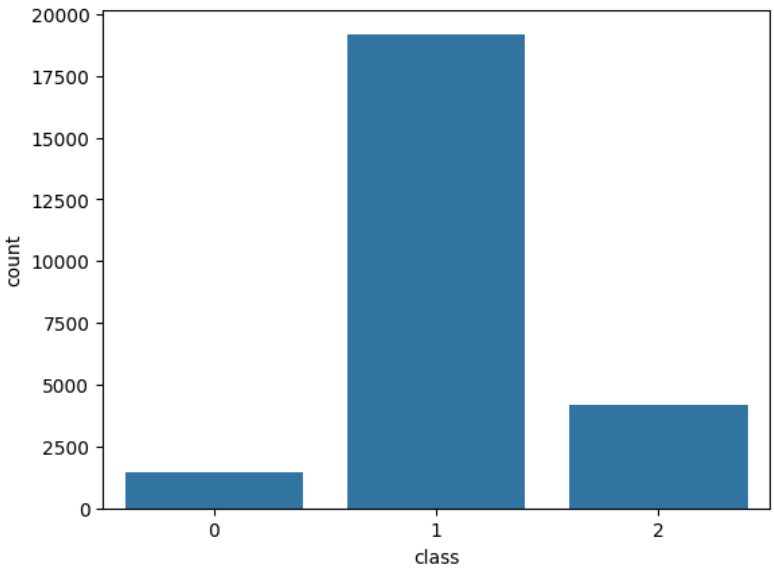
3.2 Experiments

3.2.1 Exploratory data analysis on the Davidson dataset

In order to get better acquainted with the class distribution of the tweets, we have performed some exploratory data analysis (EDA) operations on the dataset. First of all, we have seen that the dataset contains 24 783 valid tweets (i.e. after removing any duplicates and unlabeled messages). Out of these, the vast majority (22 807, to be more specific) were labeled by 3 annotators. However, there were 1571 tweets that were annotated by 6 annotators, and even 167 tweets that required 9 annotators to label them before reaching a conclusion. The following graph illustrates the distribution of the tweets with respect to the number of annotators:



As far as the class distribution is concerned, EDA has shown that 5.77% of the tweets belong to class 0 (which represents hate speech), 77.43% belong to class 1 (which represents offensive language) and 16.80% belong to class 2 (which encapsulates the tweets that are neither offensive, nor do they contain hate speech). The following graph shows in a visual manner the class distribution:



3.2.2 Distillation methods

In order to try and get smaller models that effectively detect hate speech from larger ones, we have experimented with three distillation methods:

- teacher annealing
- distillation on the feedforward network
- distillation on the feedforward network and on the CLS embeddings

It is to be noted that in the case of the last two methods, we have used Kullback-Leibler divergence on the logits as the loss function in the hidden layers of the network and mean squared error (MSE) as the loss function for the output of the feedforward part of the network, as opposed to cross-entropy loss everywhere, in the case of teacher annealing.

The teacher annealing method, described in Clark et al, 2019 [3], seeks to "blend" both the correct output (i.e. the label) and the teacher prediction into the loss function computation. For each task τ, let y_τ^i be the label of the i-th example used in that task, x_τ^i be the i-th example used in the task, θ_τ be the parameters that are learned by the teacher model during the course of its training and θ be the parameters learned by the student model during its training. In this way, $f_\tau(x_\tau^i, \theta_\tau)$ will be the prediction of the teacher model and $f_\tau(x_\tau^i, \theta)$ will be the prediction of the student model. In this case, the loss function will be a sum of terms of the form

$$l_{TA} = l(\lambda \cdot y_\tau^i + (1 - \lambda) \cdot f_\tau(x_\tau^i, \theta_\tau), f_\tau(x_\tau^i, \theta))$$

where λ is a parameter that increases linearly from 0 to 1 during the course of the training, as opposed to the usual

$$l_{reg} = l(f_\tau(x_\tau^i, \theta_\tau), f_\tau(x_\tau^i, \theta))$$

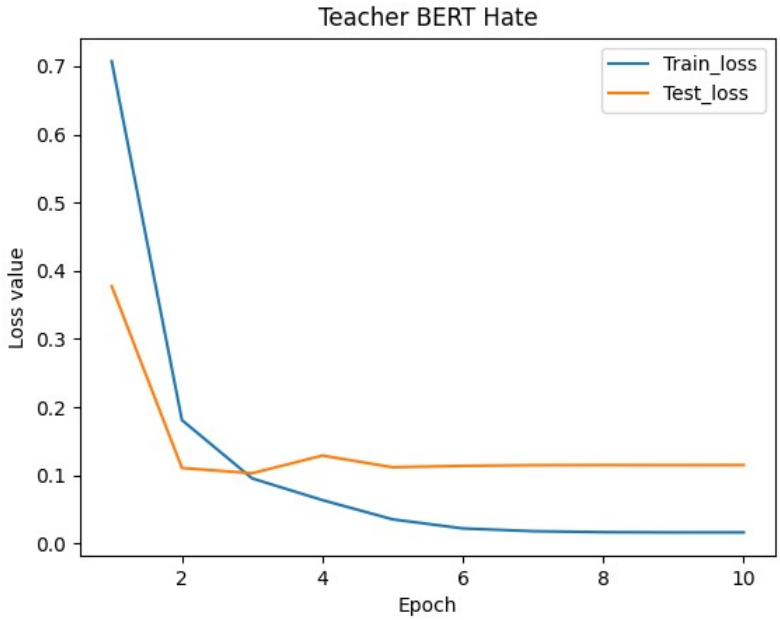
This has the advantage of allowing the student model to surpass its teacher, by almost ignoring its output and learning directly from the labels during the final stages of the training.

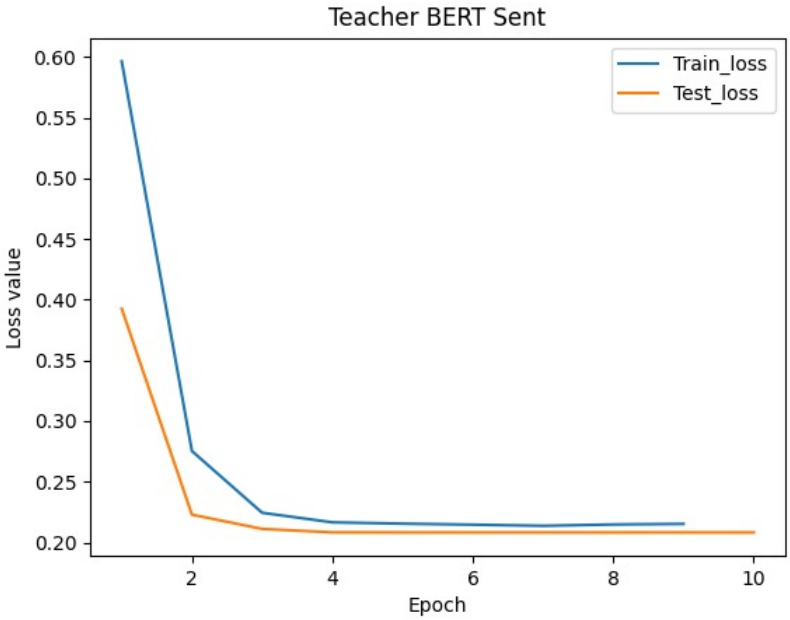
The next two methods are both described in Zhang et al, 2024 [16] and, as mentioned by the authors, the methods require using the Kullback-Leibler divergence on the token embedding and token embedding similarity matrices of the teacher and the student, respectively (for embedding layer distillation, use KL divergence on the raw token embedding matrices and for the prediction layer, use KL divergence on the

token embedding similarity matrices). Furthermore, they both use as their loss function the average over the input sequence length of the KL divergence values. The difference between the two methods is the range of tokens on which the distillation is applied onto.

4 Results

The first experiment we have performed after changing the hate speech dataset was to train the baseline teacher BERT model on the new dataset. This time, we have obtained for the teacher model an F1 score of 92% on the hate speech detection task and around 99% on the sentiment analysis task[1]. The training and test loss graphs for the two tasks at hand are the following:

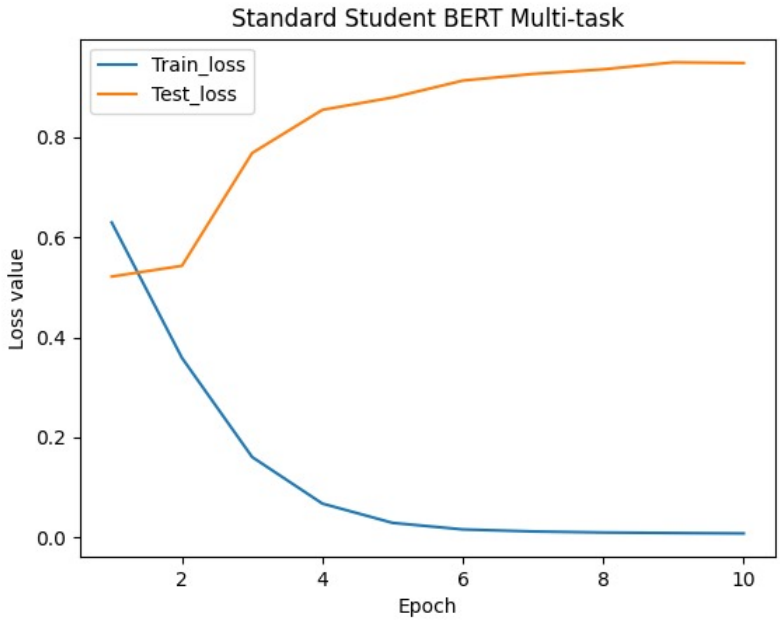




[1]All metrics were computed on the validation part of each dataset.

The graphs clearly show that the BERT model does a very good job at detecting both the degree of positivity (or negativity) of a text and whether a text represents hate speech or not. In the sentiment analysis task, the model actually performs better on the test data than on the training data for every epoch (except for the last one, for which no data is available).

We have also trained the baseline student BERT model on the two datasets, and we have obtained an F1 score of 74.89% for hate speech detection and 86.88% for sentiment analysis. The training and testing losses for this setup are provided below[2]:

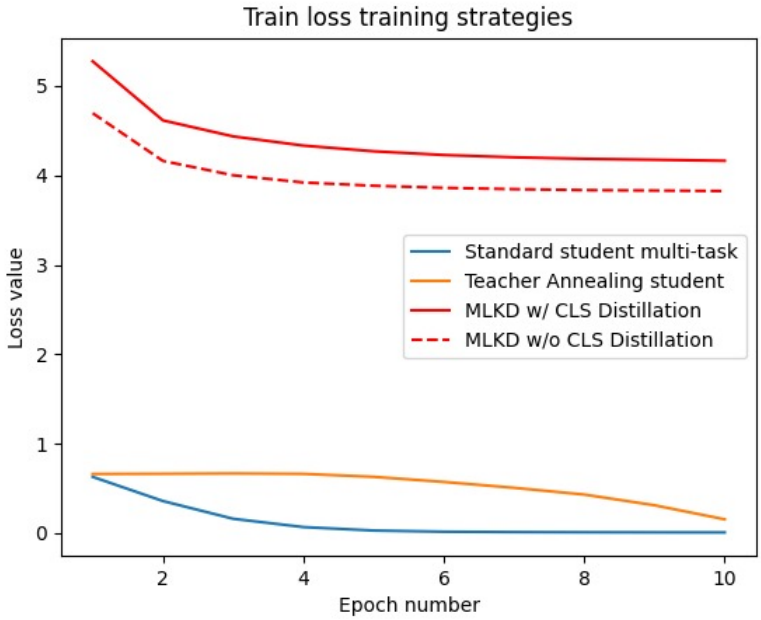


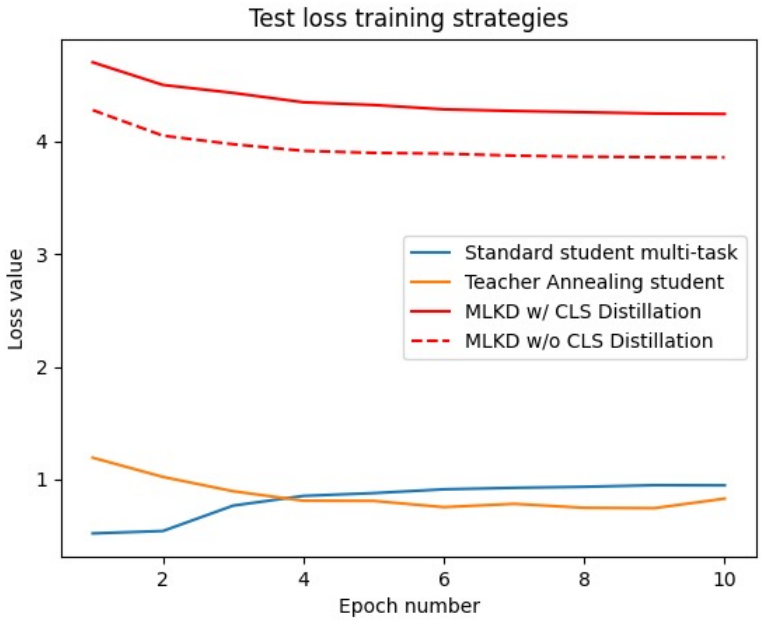
From the provided graph one can see that, despite its high F1 scores for both individual tasks, the model struggles to learn how to solve the tasks at hand (on the testing set, the loss increases to about 1, while on the training set, it almost vanishes during the 10 training epochs).

For the teacher annealing method, we have achieved an F1 score of 74.30% on the hate speech detection task and 81.92% on the sentiment analysis task. For the distillation just on the feedforward netork, we have obtained an F1 score of 72.43% on the hate speech task and 65.86% on the sentiment analysis task. The distillation both on the feedforward network and on the CLS tokens has produced almost identical results: F1 score of 72.34% on the hate speech task and 65.86% on the sentiment analysis task.

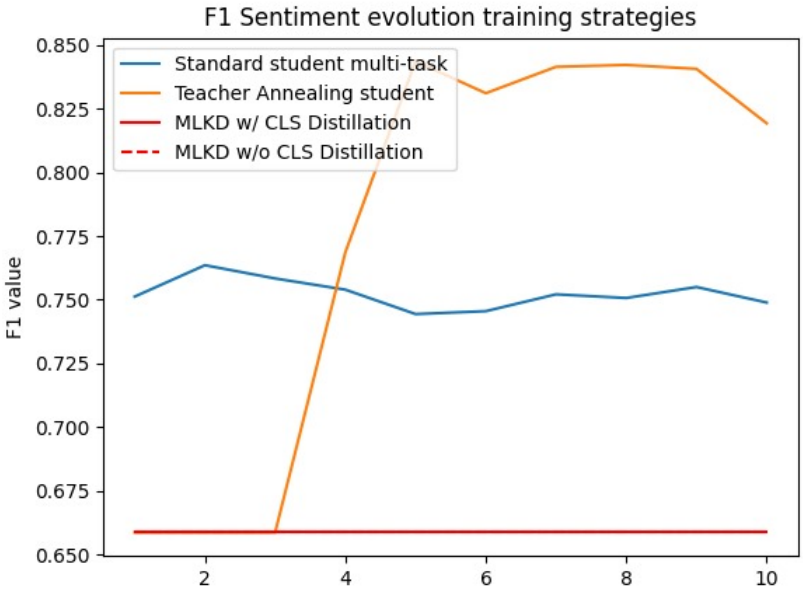
The following graphs provide a visual way of interacting with the results:

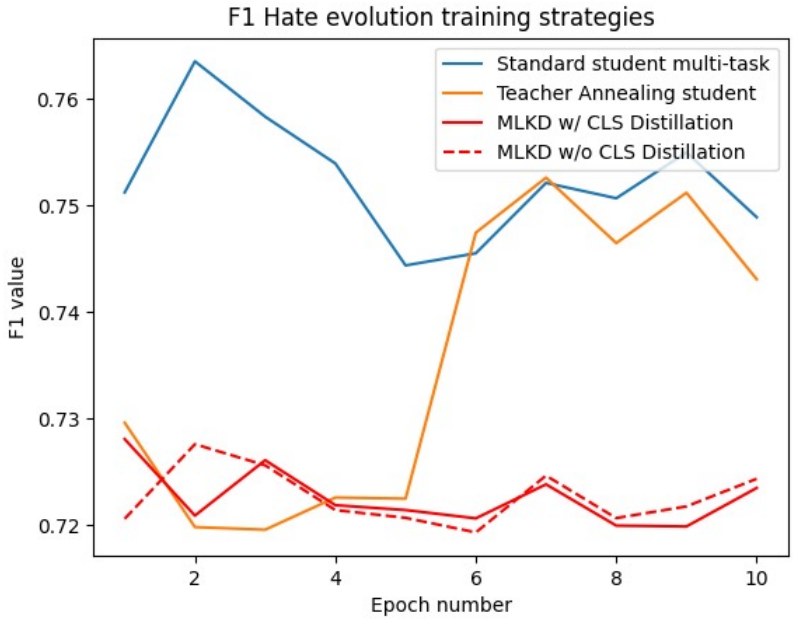
[2]We have performed the other experiments in the order in which we have described them in section 3.2.2.



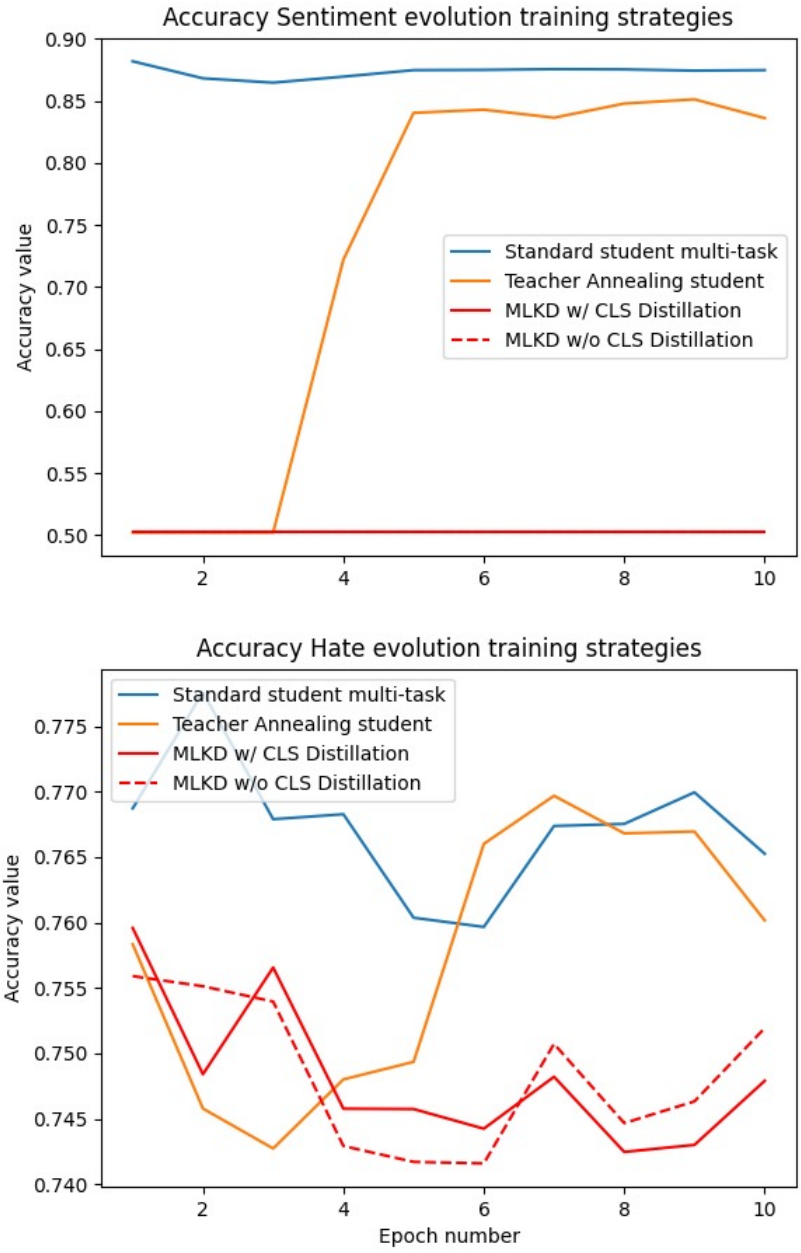


These graphs show that the models produced using the last two methods perform more poorly both in terms of training loss and of testing loss, which means that they do not do as good of a job at predicting sentiments in general, and hate speech in particular, as the baseline model and the one that uses teacher annealing. It is to be noted that the student that was trained using teacher annealing obtained very similar training and test losses with the baseline model (which are much lower than the losses achieved by the other two student models).





On the hate speech detection task, one can see that the student using teacher annealing has an F1 score which is very similar to that of the baseline student model, and they both perform visibly better than the students that use distillation inside the feedforward network. On the sentiment analysis task, it is evident that the student trained with teacher annealing has outperformed by about 7.5% the baseline student, which in turn has outperformed the other two students by about 10%. This difference in performance may have to do with the fact that the models have to learn to correctly predict 10 classes for sentiment analysis, as opposed to just 3, for hate speech detection, and this may unleash the power of the teacher annealing (the training using the correct labels near the end of each epoch).



Once again, teacher annealing has provided similar results to those of the baseline model: the accuracy on the sentiment analysis task is about 5% lower than that of the baseline model, at around 84% (which is significantly better than that of the other two models, which have obtained an accuracy of about 50%). On the hate speech detection task, there is, once again, a visible difference between the models, although not as evident as the one in the previous graph, which reinforces the performance hierarchy of the four models.

5 Conclusions

Given that Yuan and Rizoiu [15] have obtained F1 scores of 68.22% and 64.55% using an MTL-NCH and an MTL-MV model, respectively, on the Davidson dataset and that Plaza-del-Arco et al, 2021 [10] have obtained on the Hateval dataset an average F1 score of 78.47% when using the multi-task architecture that was pre-trained for both sentiment analysis and emotion detection, we consider that our best results approach the state-of-the-art, with respect to the hate speech detection task.

Furthermore, given that, for the sentiment analysis task, Csanády et al, 2024 [4] have managed to obtain an accuracy of 96.68% on the IMDb dataset using a RoBERTa model together with a LlamBERT (in their study, the F1 measure was not computed), it is visible that our two best student models, the baseline and the teacher annealing one, are relatively close to the performance in the state-of-the-art for a teacher model. Thus, we can affirm that our best student models have managed to perform well on the sentiment analysis task. Moreover, our baseline teacher model has managed to obtain an F1 score of approximately 99% on the same task and on the same dataset, which we consider to be at the same level as the current state-of-the-art.

All in all, in our opinion, this project has shown that distillation has the potential of generating student models that mimic the performances of their teachers. This has the evident benefit of being able to use smaller (thus, more computationally efficient) models in order to solve natural language processing tasks like sentiment analysis and hate speech detection without incurring much penalty performance-wise.

References

[1] Dylan Auty, Roy Miles, Benedikt Kolbeinsson, and Krystian Mikolajczyk. Learning to project for cross-task knowledge distillation, 2024.

[2] Qiwei Bi, Jian Li, Lifeng Shang, Xin Jiang, Qun Liu, and Hanfang Yang. Mtrec: Multi-task learning over bert for news recommendation. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2663–2669, 2022.

[3] Kevin Clark, Minh-Thang Luong, Urvashi Khandelwal, Christopher D. Manning, and Quoc V. Le. BAM! born-again multi-task networks for natural language understanding. In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5931–5937, Florence, Italy, July 2019. Association for Computational Linguistics.

[4] Bálint Csanády, Lajos Muzsai, Péter Vedres, Zoltán Nádasdy, and András Lukács. Llambert: Large-scale low-cost data annotation in nlp. *arXiv preprint arXiv:2403.15938*, 2024.

[5] Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language. In *Proceedings of the 11th International AAAI Conference on Web and Social Media*, ICWSM '17, pages 512–515, 2017.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

[7] Eldrich. Hate speech offensive tweets by davidson et al.

[8] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015.

[9] Abiola Obamuyide and Blair Johnston. Meta-learning adaptive knowledge distillation for efficient biomedical natural language processing. In Yulan He, Heng Ji, Sujian Li, Yang Liu, and Chua-Hui Chang, editors, *Findings of the Association for Computational Linguistics: AACL-IJCNLP 2022*, pages 131–137, Online only, November 2022. Association for Computational Linguistics.

[10] Flor Miriam Plaza-Del-Arco, M. Dolores Molina-González, L. Alfonso Ureña-López, and María Teresa Martín-Valdivia. A multi-task learning approach to hate speech detection leveraging sentiment analysis. *IEEE Access*, 9:112478–112489, 2021.

[11] Shishir Roy, Nayeem Ehtesham, Md. Saiful Islam, and Sabir Ismail. Multitask learning as question answering with bert. In *2021 24th International Conference on Computer and Information Technology (ICCIT)*, pages 1–6, 2021.

[12] Raphael Tang, Yao Lu, Linqing Liu, Lili Mou, Olga Vechtomova, and Jimmy Lin. Distilling task-specific knowledge from bert into simple neural networks, 2019.

[13] Matthew Wallingford, Hao Li, Alessandro Achille, Avinash Ravichandran, Charless Fowlkes, Rahul Bhotika, and Stefano Soatto. Task adaptive parameter sharing for multi-task learning, 2022.

[14] Qi Yang and Lin Shang. Multi-task learning with bidirectional language models for text classification. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2019.

[15] Lanqin Yuan and Marian-Andrei Rizoiu. Generalizing hate speech detection using multi-task learning: A case study of political public figures. *Computer Speech & Language*, 89:101690, 2025.

[16] Ying Zhang, Ziheng Yang, and Shufan Ji. Mlkd-bert: Multi-level knowledge distillation for pre-trained language models, 2024.