Anticipating Hate Speech from partial Input

Nevò Mirzai Hamadani **Federico Rocca** **Timothé Dard**
Group 30

Abstract

Our work proposes a model for predicting the probability of hate speech as a sentence is composed. This could enable proactive moderation before content is published or during live events. We build on transformer-based encoder and decoder models, and optionally recurrent layers (RNNs) and we evaluate performance on both standard datasets and a custom dataset we manually annotated with human-perceived hate probabilities for partial inputs.

Keywords: Hate speech, prediction, text encoders, text decoders, BERT, GPT-2, RNNs, biLSTM

1. Introduction

Hate speech is appearing more and more on social media, live-streams and comment threads, driving the need for moderation tools that intervene *while* a message is being written rather than *after* it appears. By estimating hateful intent before the "Send" button is pressed, such a system could block harmful content from ever reaching its audience.

Most previous work, however, only labels complete sentences with a binary label (hate or non-hate). Transformer encoders like BERT and its specialized variants already deliver high accuracy on full-text classification tasks [1, 2], and early-intervention studies adapt these models to drafts or replies [3, 4]. Other research on text classification designs loss functions that reward both speed and correctness [5], but it remains unclear if these approaches can produce a probability estimate at each typing step.

To understand whether a predictive model can be trained to estimate the probability of a message becoming hateful based on partial input, both decoder models (GPT-2) and encoder models (BERT) were tested. These two types offer complementary strengths: GPT-2 is designed to predict upcoming tokens, while BERT is better at classifying given text. We (i) train each model to perform hate/non-hate classification on large public datasets where sentences are cut in prefixes, and (ii) fine-tune them on a small, hand-annotated dataset where each sentence prefix is labeled with a subjective hate probability. We then experimented by extending each pipeline with RNN layers to capture sequential dependencies and introduce a weighted loss that scales inversely with prefix length, emphasizing early predictions.

Our experiments reveal a clear trade-off. Decoder-based pipelines provide better results when training on a binary dataset, while encoder-based pipelines yield the best probability scores. Incorporating RNN layers and weighted loss improves performance in some cases and worsens it in others. While our results are still limited and overall performance remains mediocre, this two-stage strategy suggests that real-time hate speech prediction might be possible with further refinements, better data, and bigger and more robust models.

2. Related Work

Traditional hate speech detection focuses on classifying complete texts. Models like HateBERT, which retrains BERT on abusive Reddit data [1], and comparisons of BERT, RoBERTa and similar encoders on full-text classification tasks [2] have demonstrated strong performance. These successes motivate the use of encoder models in our approach, while also prompting us to explore decoder architectures for early prediction [6].

More recent work shifts toward proactive detection: some studies use BERT-style models to predict harmful intent before publication [3], while others analyze drafts to anticipate toxic replies [4]. There is a growing interest in real-time moderation, which would allow to intervene earlier than traditional pipelines.

In parallel, research on "early" text classification designs loss functions that balance accuracy with earliness, pushing models to decide from minimal input [5].

Unlike these methods, we aim to produce continuous probability estimates at every typing step, tracking evolving intent over the course of a sentence. Inspired by these directions, we evaluate multiple transformer encoders and decoders, integrating RNN layers to capture sequential dependencies and fine-tuning on a hand-annotated text of prefix-level hate probabilities.

EE-559: *Deep Learning*, 2025.

3. Method

The task we would like to approach can be formulated as follows: given a prefix $x_{1:i}$ of a complete sentence $x = x_{1:n}$, express a value of probability indicating how likely the complete sentence is to contain hate speech, $P(x \text{ hateful}|x_{1:i})$. The overall approach was to utilize, train, and compare two main transformer architectures, in conjunction with a classifier, to predict the probability of hate speech using a combination of existing datasets and a manually designed dataset for incremental learning.

3.1. Model selection

We selected BERT Base [7] (∼124M of parameters) and GPT-2 Small [8] (∼110M of parameters) as our base models because they are both good enough at understanding complex language, and they are small enough not to take too much time to train. BERT is a bidirectional encoder that incorporates context from both left and right of each token, while GPT-2 is an auto-regressive decoder that predicts each next token based solely on preceding text. We chose them specifically to compare an encoder (BERT) with a decoder (GPT-2), in order to see which architecture performs better for our prediction task. The models were trained using different methodologies: (i) we used one of the pretrained text transformers and trained a classifier on top of its text embeddings and (ii) we used the same pretrained model but added a biLSTM (RNN) layer between the transformer and the classifier, to help capture how meaning evolves throughout the sentence.

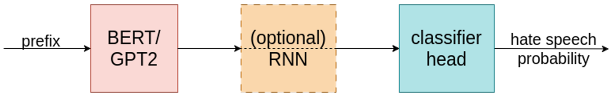


Figure 1. Flow-Chart of the model architecture

3.2. Dataset Strategy

We began with two large general hate speech datasets (around 25,000 entries), both labeled with binary hate/non-hate tags: the dataset by [9] from Hugging Face, which includes sentences from Stormfront (a white supremacist forum) labeled by researchers, and the dataset by [10], which contains offensive English tweets. These provided a broad view of hate speech. To improve detection of more subtle content, we also used the Implicit Hate dataset [11], which includes 6,347 tweets with less obvious hate.

However, these datasets only label full sentences, while our goal is to predict the risk of hate speech as a sentence is being typed. To bridge this gap, we manually created a new dataset. We selected 600 samples from the Implicit dataset [11] and broke each sentence into sequential partial phrases. For each prefix, we assigned a probability score from 0 to 1, estimating how likely the full sentence was to become hateful (Table 1).

To train our model on partial inputs, we implemented two distinct weighting strategies for the loss function.

Length-based Weighting: To account for the varying lengths of partial phrases, we incorporated a weighting factor into our loss. This weighting was determined by the ratio of the current prefix length to the length of the complete phrase ($l_{prefix}/l_{complete\_phrase}$). We applied this length-based weighting consistently across all training phases, including both the initial training on binary datasets and the fine-tuning on our hand-labeled data. This approach was chosen to reduce the penalty for incorrect predictions when only a small portion of the phrase had been observed, acknowledging the inherent difficulty of predicting hate speech from minimal context.

Label Imbalance Weighting: Most labels in the hand-labeled finetuning dataset were 0.0, resulting in a strong imbalance (Fig. 2). To address this and help the model learn from the rarer, high-probability cases, we applied another weighting factor to the loss. This specific upscaling was used only for the hand-labeled dataset. Each loss was scaled by the true label multiplied by a constant, effectively making the loss zero when the true label was 0 while increasing it when the true label was higher. This encouraged the model to focus on non-zero probability predictions. This label imbalance scaling weight is combined with the length-based weight presented above. The overall loss for the hand-labeled dataset was computed using a ratio of 0.8 for the label-based weight and 0.2 for the length-based weight.

Sentence	Label
After you strip off	0.0
After you strip off his	0.1
After you strip off his makeup	0.3
After you strip off his makeup, biologically	0.6

Table 1. Example of part of a hand-labeled sentence

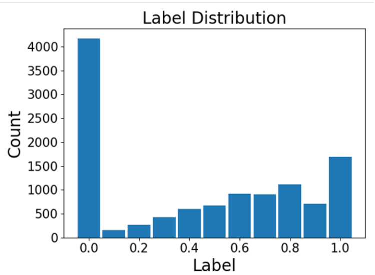


Figure 2. Labels distribution on hand-labeled dataset

3.3. Training and Evaluation

Models are first trained on the large datasets with binary labels, where phrases are cut at a random point and the loss is adjusted accordingly (Sec. 3.2). Following this, they are fine-tuned on the hand-labeled dataset. Finally, we evaluate performance on a small separate dataset created from scratch, which consists of 400 partial sentences of hate speech and their assigned probabilities of becoming hate speech, based on our subjective judgment. This comparative evaluation will help us choose the optimal model and its corresponding hyperparameters for our prediction task.

4. Validation

We split the data into 85% training and 15% validation and first trained our models on binary-labeled datasets (Sec. 3.3). During this phase, GPT-2 models, both with and without the RNN layers, showed faster improvements in F1-score and accuracy, eventually reaching about 0.78 accuracy and 0.74 F1 (Fig. 3). BERT, used alone, performed worse with around 0.66 accuracy and 0.58 F1, but adding the RNN layers slightly improved it to 0.69 accuracy and 0.62 F1 (Fig. 4). This was somewhat expected, since GPT-2 is designed to predict upcoming tokens, which fits well with our binary classification task.

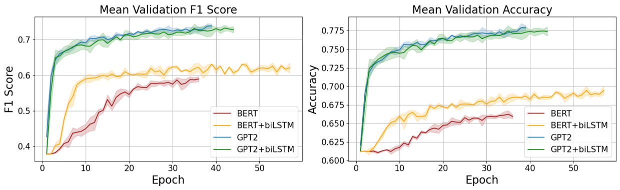
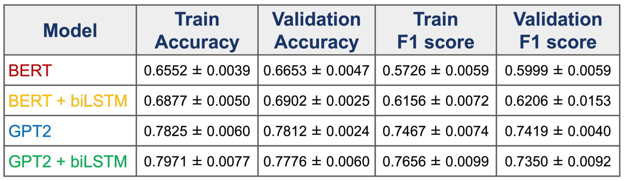


Figure 3. F1 and Accuracy of multiple models during validation



Model	Train Accuracy	Validation Accuracy	Train F1 score	Validation F1 score
BERT	0.6552 ± 0.0039	0.6653 ± 0.0047	0.5726 ± 0.0059	0.5999 ± 0.0059
BERT + biLSTM	0.6877 ± 0.0050	0.6902 ± 0.0025	0.6156 ± 0.0072	0.6206 ± 0.0153
GPT2	0.7825 ± 0.0060	0.7812 ± 0.0024	0.7467 ± 0.0074	0.7419 ± 0.0040
GPT2 + biLSTM	0.7971 ± 0.0077	0.7776 ± 0.0060	0.7656 ± 0.0099	0.7350 ± 0.0092

Figure 4. Accuracy and F1 score during Training and Validation

Interestingly, adding an RNN to BERT helped it focus more on sentence endings, encompassing the evolution of hate, making it behave more like GPT-2. This narrowed the performance gap between them during binary training. However, all models at this point had only seen yes/no labels and were not yet trained to output probabilities.

In the second phase, we fine-tuned the models on our hand-labeled dataset with probability values for each sentence prefix (Sec. 3.2). Here, the results reversed: BERT outperformed GPT-2, with the base BERT model giving the lowest MSE. This is likely due to BERT's bidirectional structure, which helps it capture subtle relationships across the sentence and makes it a good classifier for assigning soft probability labels. However, adding an RNN to BERT in this phase actually made performance worse, probably because it over-focused on the last tokens instead of the full context. Despite this, MSE at testing remained fairly high (Fig. 5), even for our best model (BERT), showing that there's still room for improvement. This could be addressed by training on larger or more diverse datasets, using bigger models, and especially by reducing label subjectivity. Since our labels are based on human judgment, a possible solution would be to have all annotators label every sentence and then average the scores to make them more consistent.

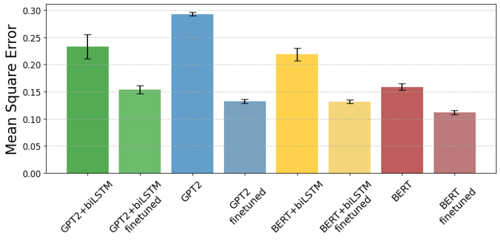


Figure 5. Testing MSE on hand-labeled data

Sentence	Probability
Immigrants bring	0.11220
Immigrants bring new ideas	0.23021
Immigrants bring new ideas and	0.17185
Immigrants bring new ideas and energy	0.20592

Sentence	Probability
I am not	0.07387
I am not sexist	0.35955
I am not sexist, but	0.48145

Table 2. Probability estimates for a neutral sentence (1st) and potentially hateful sentence (2nd)

5. Conclusion

In this work, we introduced a proactive framework for hate speech moderation that predicts the probability of hateful content from partially written text. We tested transformer-based encoders (BERT) and decoders (GPT-2), with optional RNN layers, using a two-stage training process: first on large, binary-labeled datasets, then fine-tuned on a small, hand-annotated set of incremental inputs. Our results showed that GPT-2 performs better for early binary classification, while BERT gives more accurate probability estimates for subtle, evolving hate speech. A weighted-loss strategy helped address label imbalance, improving learning from ”rare” high risk prefixes. However, accuracy remains modest, especially for ambiguous or very short prefixes, and performance drops on unfamiliar language styles, pointing to the need for more diverse and larger datasets. Future work could involve using larger models, expanding the hand-labeled data (e.g., multilingual or user-generated text), and adding real-time user feedback to refine predictions. By estimating hate probability as a message is typed, this system could intervene before publication, in live chats or comment sections, reducing harmful content and supporting safer online spaces.

References

[1] T. Caselli, V. Basile, J. Mitrović, and M. Granitzer, "Hatebert: Retraining bert for abusive language detection in english," 2021.

[2] M. S. Jahan, D. R. Beddiar, M. Oussalah, N. Arhab, and Y. Bounab, "Hate and offensive language detection using bert for english subtask a," in *Fire*, 2021.

[3] S. Bandara and H. Abeysundara, "A predictive model for anticipated hate and speech violence in social media: Large language model approach," *International Journal of Research and Scientific Innovation*, vol. XII, pp. 325–332, 03 2025.

[4] R. Alharthi, R. Alharthi, R. Shekhar, A. Jiang, and A. Zubiaga, "Will i get hate speech predicting the volume of abusive replies before posting in social media," 2025.

[5] A. Cao, J. Utke, and D. Klabjan, "A policy for early sequence classification," 2023.

[6] Isha, Anjali, K. Sharma, Kirti, and V. Pratap, "Classifying toxic comments with machine learning and deep learning approaches," *International Journal of Scientific Research in Science and Technology*, vol. 12, pp. 1073–1082, 04 2025.

[7] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[8] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI*, 2019. https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf.

[9] O. de Gibert, N. Perez, A. Garcia-Pablos, and M. Cuadros, "Hate Speech Dataset from a White Supremacy Forum," in *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*, (Brussels, Belgium), pp. 11–20, Association for Computational Linguistics, Oct. 2018.

[10] M. Zampieri, S. Malmasi, P. Nakov, S. Rosenthal, N. Farra, and R. Kumar, "Predicting the type and target of offensive posts in social media," 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)* (J. Burstein, C. Doran, and T. Solorio, eds.), (Minneapolis, Minnesota), pp. 1415–1420, Association for Computational Linguistics, June 2019.

[11] M. ElSherief, C. Ziems, D. Muchlinski, V. Anupindi, J. Seybolt, M. De Choudhury, and D. Yang, "Latent hatred: A benchmark for understanding implicit hate speech," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2021.