Modern Approaches for Ad Hominem Detection in Social Media

Diana-Nicoleta Grigore, Ioana Pintilie, coordinator: Florin Brad*

University of Bucharest, Romania
*Bitdefender, Romania

1. Introduction

When having debates on social media, users will often resort to logical fallacies, such as strawman, slippery slope, circular reasoning etc. In this paper we tackle the problem of detecting ad hominem attacks. We train several strong models (BERT/Bidirectional LSTM) with different strategies for dealing with the class imbalance. To the best of our knowledge, we obtain the best results on the ad hominem dataset by (Habernal et al., 2018) by finetuning a BERT using a weighted cost

2. Dataset

We use an ad hominem dataset which was initially collected from Reddit for a context-centered analysis [1] and then was transformed to fit an ad hominem detection scenario [2] by keeping individual posts as examples instead of the whole discussion tree. The training dataset has 23374 training examples, of which approximately 10 percent are ad hominem.

3. Class imbalance

To combat the imbalance issue, we used a weighted cross-entropy loss function for all the models. Specifically, we set the weight of the normal examples to 1, and we increase the importance of the ad hominem examples by rescaling their loss with weight $p \in \{3, 5, 7, 10\}$. We also try to oversample the minority class by using an imbalanced dataset sampler

4. Recurrent Models

Our first approach was a Long Short-Term Memory model. We tokenize the examples using a Reddit tokenizer, which conserves some of the online conversation context by treating usernames and subreddit mentions as distinct tokens. We use Word2vec embeddings to initialize the embedding layer. We apply a dropout of 0.5 on the the final hidden state of LSTM, which we next feed to an MLP with hidden size of 256. We also train a bidirectional LSTM with maxpooling over time. Both recurrent models were optimized using Adam with learning rate 1e-3.

5. BERT

We then fine-tune BERT on our dataset. To classify, we add a dropout of 0.1 over the h[CLS] sentence representation and then add a linear layer on top of it. We truncate the examples to a maximum of 64 tokens and train with batches of 128 examples. We only fine-tune the linear layer and bias terms in the BERT backbone. We also tried to fine-tune the whole model as well as the linear layer only, but with worse results. We use the AdamW optimizer with best learning rate lr=1e-3.

References

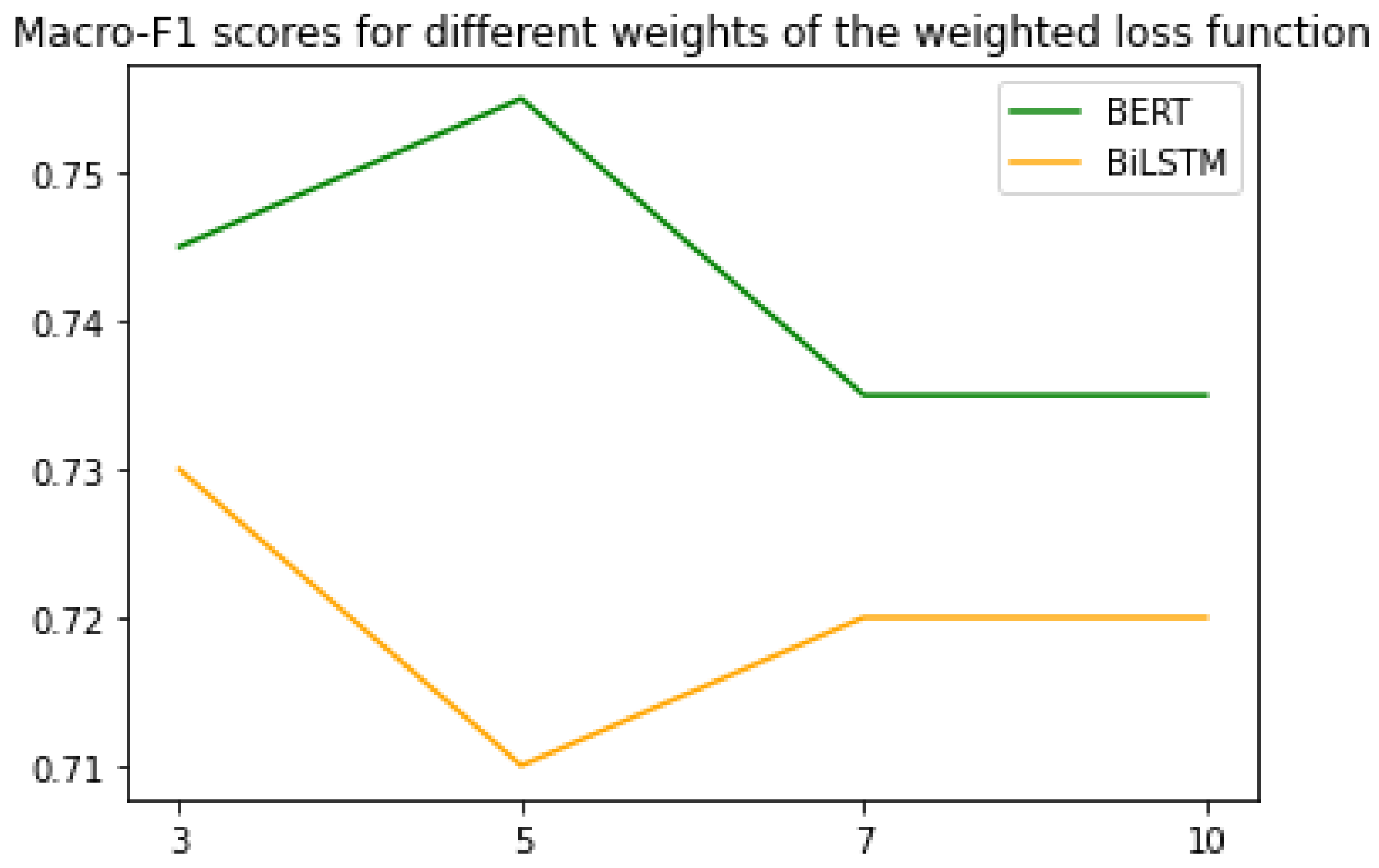
[1] Ivan Habernal, Henning Wachsmuth, Iryna Gurevych, and Benno Stein. Before name-calling: Dynamics and triggers of ad hominem fallacies in web argumentation. *arXiv preprint arXiv:1802.06613*, 2018.

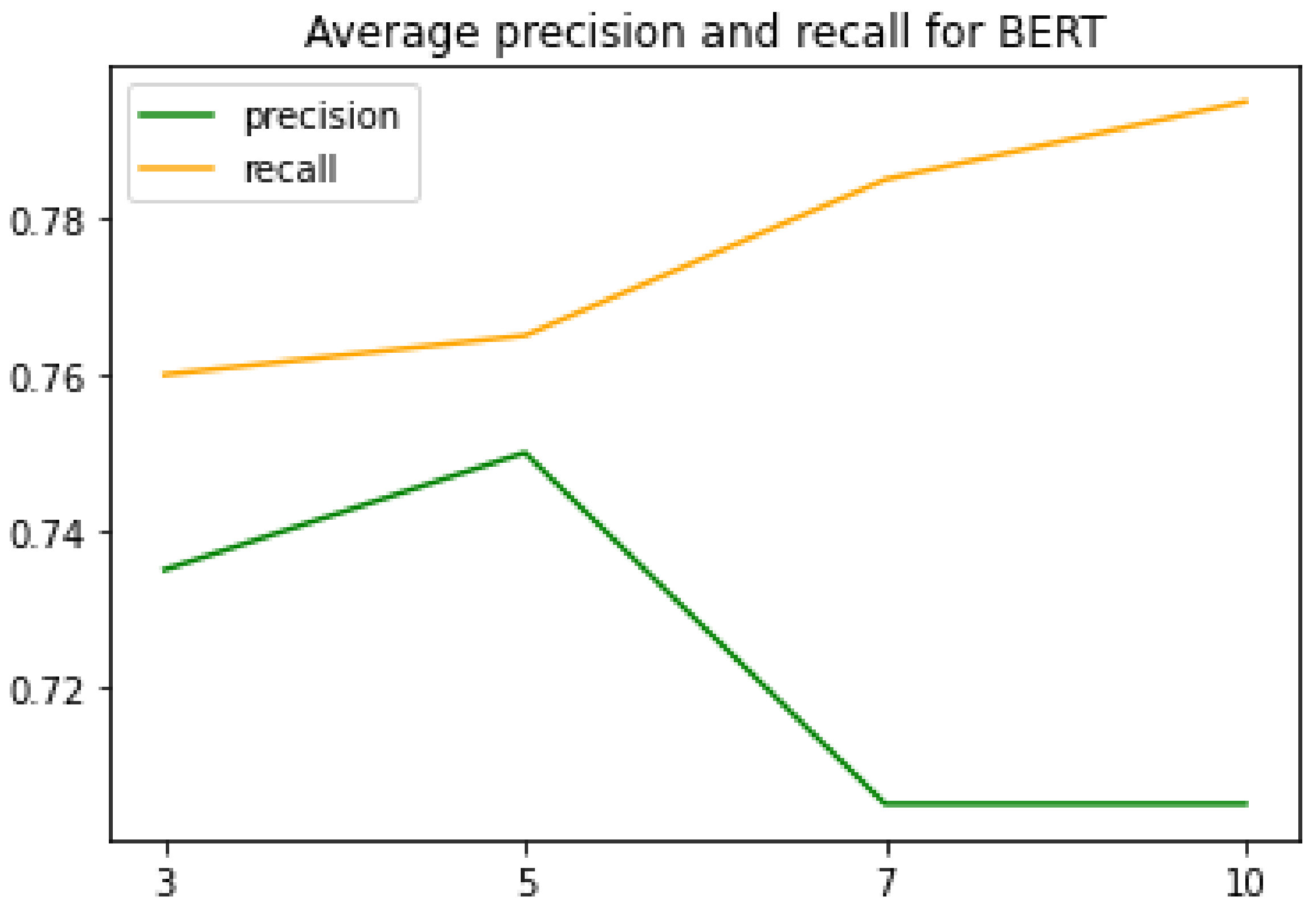
[2] Pieter Delobelle, Murilo Cunha, Eric Massip Cano, Jeroen Peperkamp, and Bettina Berendt. Computational ad hominem detection. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 203–209, 2019.

6. Results

We report precision, recall and f1-score on each class, as well as the macro-averaged precision, recall and f1-score.
The BiLSTM (p=3) results in a higher macro f1-score than the LSTM. The position of the attack in a paragraph correlates with the model's confidence in detecting ad hominem [2]. This could explain the better results of the BiLSTM model, which better integrates information from the beginning of the sentence. BERT consistently outperforms the BiLSTM model on both class imbalance strategies. Oversampling is competitive with the weighted cost approach, but choosing the appropriate weight factor for the minority class leads to the best f1 scores for both models.

Model	Prc-0	Prc-1	Rec-0	Rec-1	F1-0	F1-1	Av.g prec	Avg. rec	Avg. F1
BERT p=3	0.96	0.51	0.95	0.57	0.95	0.24	0.735	0.76	0.745
BERT p=5	0.96	0.54	0.95	0.58	0.95	0.56	0.75	0.765	0.755
BERT p=7	0.96	0.45	0.92	0.65	0.94	0.53	0.705	0.785	0.735
BERT p=10	0.97	0.44	0.91	0.68	0.94	0.53	0.705	0.795	0.735
BERT samp	0.96	0.47	0.93	0.65	0.95	0.54	0.715	0.79	0.745
BiLSTM p=3	0.96	0.46	0.93	0.59	0.95	0.52	0.71	0.76	0.73
BiLSTM p=5	0.96	0.41	0.92	0.59	0.94	0.49	0.69	0.75	0.71
BiLSTM p=7	0.96	0.41	0.91	0.64	0.94	0.5	0.69	0.78	0.72
BiLSTM p=10	0.97	0.4	0.9	0.68	0.93	0.51	0.68	0.79	0.72
BiLSTM samp	0.96	0.41	0.92	0.58	0.94	0.48	0.69	0.75	0.71
LSTM p=3	0.96	0.44	0.93	0.54	0.95	0.49	0.7	0.74	0.72





The highest f1-scores for both models are obtained for lower p weights (p=3 for BiLSTM and p=5 for BERT) and further weighting the minority class degrades the performance. We also notice that precision and recall for BERT are inversely correlated, except when p=5 for which we obtain maximum precision (as well as maximum f1).

7. Qualitative Analysis

There is a strong correlation between the model's confidence in predicting a positive label and the use of vulgarity or outright insults. A consequence of this behaviour is some misclassification of posts making use of this type of language as containing an ad hominem attack, even when the debate opponent is not the one being directly targeted by the original poster.

False positives	**False negatives**
Not every single person making noises with their cars or motorcycles are inconsiderate a**h*le.	this post gave me aids
I'm Grim, and here's your chance to tell me why I'm an idiot.	Your username fits you.

8. Conclusions

We tackle the ad hominem problem using RNNs and BERT models and we obtain the best results to date on the ad hominem dataset. Weighted cost outperforms oversampling when choosing an appropriate rescale value for the minority class.
Qualitative analysis reveals that language in misclassified examples is more toned down.