Hashing it Out: Predicting Unhealthy Conversations on Twitter

Steven Leung
UC Berkeley
stevendleung@berkeley.edu

Filippos Papapolyzos
UC Berkeley
philpapapolyzos@berkeley.edu

Abstract

Personal attacks in the context of social media conversations often lead to fast-paced derailment, leading to even more harmful exchanges being made. State-of-the-art systems for the detection of such conversational derailment often make use of Deep Learning approaches for prediction purposes. In this paper, we show that an Attention-based BERT architecture, pre-trained on a large Twitter corpus and fine-tuned on our task, is efficient and effective in making such predictions. This model shows clear advantages in performance to the existing LSTM model we use as a baseline. Additionally, we show that this impressive performance can be attained through fine-tuning on a relatively small, novel dataset, particularly after mitigating overfitting issues through synthetic oversampling techniques. By introducing the first transformer based model for forecasting conversational events on Twitter, this work lays the foundation for a practical tool to encourage better interactions on one of the world's most ubiquitous social media platforms.

1 Introduction

Social media has become one of the primary stages where peer-to-peer discussions take place, spanning anything from everyday local matters to high level philosophical topics. A very frequent phenomenon that is noticed in such conversations is their rapid deterioration upon the presence of a personal attack, often leading to personal threats and insults which heavily undermine the level of discussion. In order to battle this issue, Social Media companies, such as Twitter and Facebook, have come up with systems that make use of human moderators who review content flagged by users. In the majority of cases, content that goes against the company's policy is removed [13].

The way this moderation tactic is set up presents a series of challenges and limitations, the most obvious being the cost of moderation services. Secondly, there have been cases of unreliable moderation in Twitter, with users receiving temporary bans simply for mentioning some specific word in their tweets. Lastly, it has to be appreciated that moderation can be quite an emotionally taxing job as content under review will often be very disturbing.

The most basic limitation of this approach, however, is that moderation only happens a posteriori and solely after users choose to report the abusive content. This leaves a significant space in time in which a conversation can escalate further and users might become the victims or perpetrators of personal attack and/or verbal abuse. To the best of our knowledge, no effort is done on the part of Twitter in order to prevent such harmful content rather than moderate it. We trust that if Social Media companies chose to assume a preventative strategy to moderation, this would reduce the instances of personal attacks and improve the overall content quality.

There are a series of issues that make detecting potentially harmful language a challenging topic. As mentioned in Twitter's enforcement policy [13], context is of utmost importance; the use of specific language might have a different interpretation when the preceding conversation is better examined. Twitter's policy mentions that factors such as whether "the behavior is directed at an individual, group, or protected category of people" and "the severity of the violation" are also considered prior to taking action.

The aim of this project is to provide a Deep Learning approach to content moderation, by means of an Attention-based Neural Network model that predicts whether a Twitter conversation is likely to deteriorate and lead to a personal attack. We believe that this will not only provide

an advantage to moderators but could also potentially be utilized to give warnings to users when they are about to tweet something which may lead to escalation.

2 Related Work

A lot of our inspiration for this project was drawn from the 2018 paper Conversations Gone Awry: Detecting Early Signs of Conversational Failure (Zhang et al., 2018) [17]. Specifically, in this paper the authors try to identify predictors of conversational derailment and make predictions using a variety of strategies. An interesting insight from the paper is that conversations containing an attacker-initiated direct question are most likely to derail. The authors also claim that human-based predictions of conversational derailment are at an accuracy level of 72%.

Past work has also been done on forecasting hostility on a dataset of Instagram comments (Liu et al., 2018) [6], analyzing features of personal attack in Wikipedia comments (Wulczyn et al., 2017) [16] and the use of attention models to improve the performance of RNN and CNN approaches to content moderation (Pavlopoulos et al., 2017) [8].

In the 2019 paper titled Trouble on the Horizon: Forecasting the Derailment of Online Conversations as they Develop, Chang et al. [1], the authors use a Recurrent Neural Network (RNN) based approach to predict conversational derailment on a dataset of Wikipedia talk page discussions developed as part of the aforementioned 2018 paper, and a dataset of conversations in the ChangeMyView subreddit. CRAFT, their model architecture, yields an accuracy score of 66.5% which, to the best of our knowledge, is the highest accuracy score achieved on this specific task.

Despite the success of the RNN-based CRAFT model, we choose to explore a pre-trained self-attention model for our conversational task for several reasons. First, work by Vaswani et al. [14], and others have shown performance advantages of transformers vs RNNs. Namely, RNNs have displayed challenges with long range dependencies and word sense disambiguation relative to transformers [12]. We deemed that using an Attention model could allow us to curb the sequential architecture of RNNs, allowing the model to make use of previous context without information loss. This is grounded in the idea that previous tweets in a conversation may have an equally strong effect on the probability of derailment of a subsequent tweet. In addition, transfer learning allows us to take advantage of pre-trained models, such as BERTweet, which is trained on millions of examples in the unique language of Twitter, thus improving the overall performance of our approach. Finally, due to the linear scaling self-attention models as opposed to the quadratic scaling of RNNs, our model is more scalable to the vast conversational world of Twitter.

3 Dataset

For this project we have compiled our own dataset, consisting of 5656 tweets tagged on one binary dimension of personal attack. Tweets are labelled with conversation ID's, vital for our task of using prior tweets to predict the outcome of a subsequent tweet. We acquired our data through the Twitter API. In order to have a higher probability of finding positive examples of personal attack, we selected conversations from a collection of controversial topics such as Universal Basic Income, abortion, and immigration.

In tagging each tweet as either containing a personal attack or not, we used the Wiktionary definition of a personal attack: "an abusive remark on or relating to somebody's person instead of providing evidence when examining another person's claims or comments". We used three methods in tagging. First, we made use of the participant recruitment platform Mechanical Turk. Secondly, we were fortunate enough to find a dataset on Zenodo titled "Hate speech and personal attack dataset in English social media" [2] that contained tweet level tags. We were able to match these tweets to the rest of the conversation through the Twitter API. Finally, we tagged the remainder of tweets ourselves.

From the 5656 tweets, we isolated 1177 positive examples, leaving 4479 negative examples which translated to a heavy class imbalance. Our budget and project time frame were significant limitations to the quality and scale of our data collection, forcing us to seek alternative methods of reducing the effect of positive example under-representation, which came at their own costs. Specifically, we made use of oversampling for positive examples, which came at the expense of overfitting.

3.1 Synthetic Oversampling

A method we made use of to reduce class imbalance in our dataset was a technique called synthetic

oversampling. Specifically, we deemed that by creating artificial context examples for the positive class and supplying them to our model in the training phase we could reduce the effect of overfitting while also providing a larger quantity of plausible training examples. This technique works by replacing words in the original context with similar words to create new contexts and resupplying them to the model as separate training examples.

Our inspiration was drawn from the 2016 paper A Study of Synthetic Oversampling for Twitter Imbalanced Sentiment Analysis (Ah-Pine & Soriano, 2016) [9] in which the authors present a general algorithm for creating synthetic samples as follows: a random tweet x is drawn from a distribution P, its nearest neighbors NN(x) are determined, a neighbor x' is chosen "according to a probability distribution over NN(x)" and a synthetic sample is created in the form $y = x + a(x' - x)$, where $a\ unif(0, 1)$. A popular version of this algorithm is known as SMOTE (Synthetic Minority Oversampling TEchnique) (Chawla, 2002) [3], which assumes a uniform distribution over P. The application of this algorithm in the context of words requires a vector space mapping which, in our case, was achieved using the GloVe Twitter Embeddings.

Specifically, the original context is first broken down into a series of tokens which are tagged by part-of-speech and are filtered for stopwords, which are not further processed. Using the GloVe Twitter embeddings, a dictionary of k-nearest neighbors is computed for each token using the Euclidean distance metric. We randomly pick one of the 3 closest neighbors with non-uniform probability, giving larger probability to the closest neighbor. We then randomly choose the filtered tokens to be replaced in the original context with probability P = 0.2 and repeat this process n times to produce a list of similar but non-identical contexts. Using this process we developed 355 synthetic positive examples.

4 Transformer Model

Our general model for forecasting conversational events is the BERT-base BERTweet transformer model, fine-tuned on our conversational Twitter data.

4.1 Conversational Structure

Conversations For the purposes of our model, a conversation is made up of an initial tweet (what we call a top-level tweet) and all subsequent tweets following that tweet that are in direct reply to one another. While Twitter does allow for the branching of conversations (at any point in a conversation, users can reply directly to the top-level tweet or to a reply, a reply of a reply, etc.) that allows for complex conversational structures, we limit our model to looking at single conversational branches at a time for each input.

Thus, given top-level tweet T1, a conversation consists of a sequence of N tweets $C = \{T_1, ..., T_N\}$

Context Tweets are the individual components of the conversation, made by a single user. For the sake of forecasting, we are using the two tweets prior to forecast whether the current tweet in question will be a personal attack. The two tweets prior are referred to as the "context". This is the input into our model.

The two context tweets are concatenated with the $< /s >$ token in between to preserve the delineation of tweets. The context is then tokenized using the BERTweet tokenizer, with a normalization feature to handle common conversational elements in Twitter such as username handles, hashtags and urls. The classification token, $[CLS]$, is appended to the front of the input for use in the classification process. Thus, given the tweet in question, T_N, the two tweet context, $T_N - 1$ and $T_N - 2$, is converted to variable M length tokens, yielding the conversational context, $C_N = [CLS], T_{N-1,1}, T_{N-1,2}, T_{N-1,...}, < /s >, T_{N-2,1}, T_{N-2,2}, T_{N-2,...}$.

The most recent tweet in the context is placed in the front of the context to avoid truncating the most recent tweet in the case that the context exceed BERTweet tokenizer's 130 token length limit.

4.2 Fine-tuning BERTweet

The pre-trained BERTweet model that we use is a fine-tuned version of the popular BERT model. At a high-level, BERT is a bi-directional encoder model trained to predict masked tokens and whether one sentence directly follows another in the source text [4]. We use the base model configuration consisting of 12 transformer blocks, 12 self-attention heads, and a hidden size of 768. In the BERTweet implementation, BERT base is fine-tuned on 850M tweets for optimal performance on Twitter data in regards to the tasks of part-of-speech tagging, named-entity recognition and text classification [7]. We choose this pre-trained

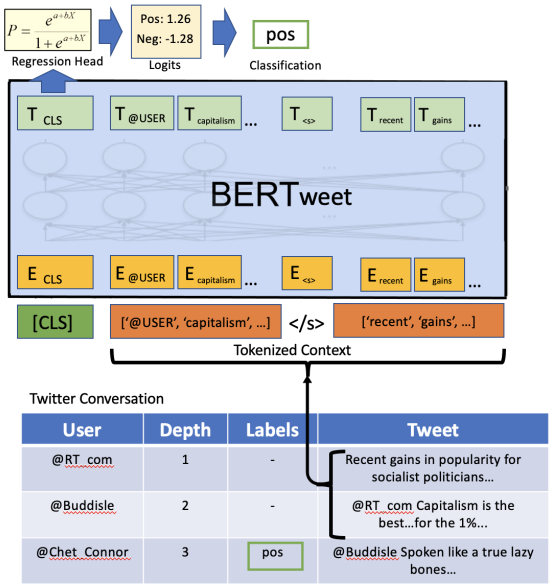

Figure 1: Model architecture. The tokenized two tweet context is fed into pre-trained BERTweet. Forecasting whether the subsequent tweet will be a personal attack or not is treated as a classification problem through a logistic regression head on the CLS token final embedding.

model based on its superior performance to other transformer models such RobertaLarge and XLM-R large, specifically on Twitter data on the tasks mentioned above.

During the fine-tuning process, we update the parameters of BERTweet on the specific task of personal attack forecasting using Twitter conversation context in accordance with the method proposed in Howard et al. [5]. We append a classification head on top of BERT using the huggingfaces model configuation Bertforsequenceclassification [15]. In this implementation, the [cls] token is used to classify whether the tweet in question is forecasted to be a personal attack or not, in that the final hidden state h of this token is used to classify the sequence. A simple logistic regression classification head is added to the top of BERTweet to forecast the probability of a personal attack:

$$P(\text{Attack} \mid H_{1...768}) = \text{Softmax}(WH_{1...768}) \quad (1)$$

This classification method is discussed in further depth by Sun et al. [11].

5 Evaluation and Analysis

5.1 Results

We test and evaluate three models to examine the effect of both the model architecture and synthetic oversampling. We use the CRAFT model as our baseline in the same form as proposed by Chang et al.. CRAFT is trained and evaluated on our Twitter conversation data using standard oversampling on the positive class. We then fine-tune and evaluate BERTweet with standard oversampling to measure the impact of a transformer implementation as opposed to the LSTM implementation proposed in CRAFT. Finally, we fine-tune and evaluate BERTweet on our Twitter conversation data with synthetic oversampling of the positive class.

Model	A	P	R	F1	AUPR
CRAFT	0.76	0.52	0.57	0.54	0.55
BT	0.82	0.62	**0.76**	0.68	0.76
BT SOS	**0.85**	**0.69**	0.72	**0.70**	**0.78**

Table 1: Comparison of performance between CRAFT with random oversampling (CRAFT), BERTweet fine-tuned with random oversampling (BT) and BERTweet fine-tuned with synthetic oversampling (BT SOS). Classification threshold of .5.

We evaluate our model in relation to several key metrics - including total accuracy on both positive and negative classes and precision, recall, F1 score and area under the precision recall curve for the positive class. While it is of vital importance to flag as many upcoming personal attacks as possible (recall), it is also of crucial importance to maintain credibility by not flagging innocuous conversations (precision). Thus, we pay particular attention to the area under the precision recall curve (AUPR) as a measure of success. The AUPR metric was used heavily in the 2019 paper by Chang and Danescu-Niculescu-Mizil, in which the authors achieved an AUPR of .70 on their final model [1], although we do not make an explicit comparison to this result due to the disparate nature of our conversational data.

The transformer based BERTweet model outperforms the LSTM based CRAFT model in all metrics. Implementing synthetic oversampling also improves all metrics, with the exception of recall on the positive class on the classification threshold of .5. Given the higher AUPR score, however, we see that synthetic oversampling has generally improved our model's ability to classify the positive class.

5.2 Analysis

We now examine the behavior of our model to better understand what aspects of the data it is using

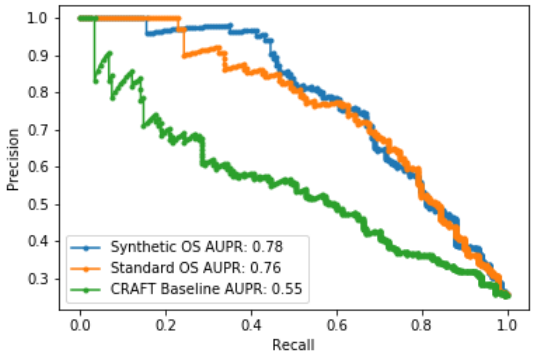

Figure 2: Precision-recall curves and the area under each curve.

for predictions and where it is falling short.

5.2.1 Model Behavior

The Lead Up to an Attack We first examine whether the model is using the full two tweet context we are supplying to make its predictions or if a single tweet context is sufficient. Intuitively, we expected only a minor drop in performance since our assumption was that most of the signal indicating an attack was coming would be captured in the tweet immediately preceding the attack tweet. We tested this hypothesis by truncating the training data to a single tweet. We were surprised to see a dramatic drop in performance, with the model performing at .74 accuracy and only .50 AUPR. This indicates that the tweet two prior does indeed provide a large amount of useful information in making predictions.

Conversational Dynamics Secondly, we look at whether the model is using information about conversational dynamics in the context to inform its predictions. To look at this, we remove the tweet separator token, $< /s >$, so that any indication of delineation between speakers is lost. Model performance again drops, although not as significantly as in our first test, with an accuracy of .77 and AUPR of .76. This meets our expectation since the location of inflammatory language the leads to a personal attack relative to the token (in other words, which tweet in the context it is in) would presumably be valuable information for the model to use in making a prediction on the current tweet.

5.2.2 Model Limitations

Limited Dataset While building our model, one of the major challenges we faced was the relatively small size of the dataset we were fine-tuning on, particularly in the positive class. The main consequence of this was a tendency of our model to overfit the training dataset. We dealt with this issue through a variety of methods, including synthetic oversampling (as mentioned earlier), reducing batch size to 10, maintaining the learning rate at the default level of 5e-5 and ensuring that training did not exceed 4 epochs.

Even so, we occasionally see unusual spikes in the negative log likelihood loss pattern on the validation set related to our limited dataset. These spikes would occur despite increasing accuracy and AUPR score. In other words, our model is making more correct predictions on the positive and negative class on our validation data but is also more over-confident in the bad predictions it is making. We posit two potential explanation for this. Firstly, despite the measures above, our model is likely still overfitting to an extent on the training data. This leads to over-confident incorrect predictions due to random noise as opposed to true signal in our validation data. Secondly, due to the small number of positive examples in our validation set (154), a small change in parameters between epochs could result in dramatic swings in total loss. In other words, a few very bad predictions could have a large impact on the loss calculation, resulting in the spike we witnessed.

GIF Confusion We observe a higher occurrence of url strings in the context strings where our model makes misclassifications on the test set. The ratio of url strings to context strings is .65 in the misclassified examples vs only .5 in the correctly classified examples. This makes sense intuitively since these url strings contain a large amount of information relative to whether a personal attack is coming that is not interpretable by our model. The best example of this is GIFs, or animated images. While these GIFs are represented in our data as simple url strings, the actual content of the GIF could be highly inflammatory, benign or even friendly, which could be highly indicative of the presence or absence of a personal attack in the subsequent tweet. While we did not have time to explore this further, we believe this would be a fruitful area of future work, as we will discuss in more detail in the next section.

Nuanced Langugage In their paper, Price et al. note the difficulty for neural models to identify

nuanced language indicating negative interactions such as sarcasm, dismissiveness, and condescension. [10]. Notably, BERT was shown by the authors to be particularly poor at identifying sarcasm. These are language qualities that could be highly indicative of impending personal attacks. Since this is not currently accounted for in our model, we believe this is likely a source of confusion contributing to our existing loss.

6 Conclusions and Future Work

In summary, we introduced a transformer based model for forecasting conversational events on a novel dataset of Twitter conversations. This model indicates some ability to understand conversational dynamics between speakers. It fills a void in the existing literature in that it provides state-of-the-art predictive performance on Twitter, a platform that has not been studied in the space of conversational forecasting.

Given Twitter's stated goal of having healthier conversations on its platform, we hope this study is a foundation for future work specific to this ubiquitous channel of communications. One vital area of expansion will be to incorporate additional topics of conversation into the model. While we focused our study on controversial political and societal topics, additional work is needed to ensure the model performs robustly on more mundane topics, since the signals leading towards a personal attack could be very different for these conversations.

While our model performed impressively in regards to our test data, we are confident that future work could further improve performance and robustness of the model. Our model could be improved by addressing all the sources of loss mentioned earlier. Giving the model a sense of nuanced language by appending an attention head for detecting the 6 attributes of unhealthy conversations as noted in Price et al. would do well to address this deficiency. Another clear area of improvement, as referenced earlier, would be to communicate the sentiment of GIFs as input to the model. Finally, the robustness of the model would be greatly improved by anyone with the resources to reliably collect and accurately tag additional tweets.

As the model becomes more accurate and robust, we hope it can become a practical tool to assist Twitter users to interact with more civility and awareness. Given this fine-tuned ability to recognize conversations headed for derailment, Twitter could proactively warn users. We believe that this awareness would allow users to guide themselves towards a more productive resolution, allowing all parties involved to have a better, more positive experience.

Figure 3: A mock-up of a conversation warning notification on a sample conversation.

7 References

References

[1] Jonathan P. Chang and Cristian Danescu-Niculescu-Mizil. *Trouble on the Horizon: Forecasting the Derailment of Online Conversations as they Develop*. 2019. arXiv: 1909.01362 [cs.CL].

[2] Polychronis Charitidis et al. *Hate speech and personal attack dataset in English social media*. Zenodo, Oct. 2019. DOI: 10.5281/zenodo.3520152. URL: https://doi.org/10.5281/zenodo.3520152.

[3] N. V. Chawla et al. "SMOTE: Synthetic Minority Over-sampling Technique". In: *Journal of Artificial Intelligence Research* 16 (June 2002), pp. 321–357. ISSN: 1076-9757. DOI: 10.1613/jair.953. URL: http://dx.doi.org/10.1613/jair.953.

[4] Jacob Devlin et al. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. 2019. arXiv: 1810.04805 [cs.CL].

[5] Jeremy Howard and Sebastian Ruder. *Universal Language Model Fine-tuning for Text Classification*. 2018. arXiv: 1801.06146 [cs.CL].

[6] Ping Liu et al. *Forecasting the presence and intensity of hostility on Instagram using linguistic and social features*. 2018. arXiv: 1804.06759 [cs.CL].

[7] Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. "BERTweet: A pre-trained language model for English Tweets". In: (2020). arXiv: 2005.10200 [cs.CL].

[8] John Pavlopoulos, Prodromos Malakasiotis, and Ion Androutsopoulos. "Deeper Attention to Abusive User Content Moderation". In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, Sept. 2017, pp. 1125–1135. DOI: 10.18653/v1/D17-1117. URL: https://www.aclweb.org/anthology/D17-1117.

[9] Julien Ah-Pine and Edmundo-Pavel Soriano-Morales. "A Study of Synthetic Oversampling for Twitter Imbalanced Sentiment Analysis". In: *Workshop on Interactions between Data Mining and Natural Language Processing (DMNLP 2016)*. Proceedings of the Workshop on Interactions between Data Mining and Natural Language Processing. Riva del Garda, Italy, Sept. 2016. URL: https://hal.archives-ouvertes.fr/hal-01504684.

[10] Ilan Price et al. *Six Attributes of Unhealthy Conversation*. 2020. arXiv: 2010.07410 [cs.CL].

[11] Chi Sun et al. *How to Fine-Tune BERT for Text Classification?* 2020. arXiv: 1905.05583 [cs.CL].

[12] Gongbo Tang et al. *Why Self-Attention? A Targeted Evaluation of Neural Machine Translation Architectures*. 2018. arXiv: 1808.08946 [cs.CL].

[13] Twitter. *Our approach to policy development and enforcement philosophy*. URL: https://help.twitter.com/en/rules-and-policies/enforcement-philosophy.

[14] Ashish Vaswani et al. "Attention Is All You Need". In: *CoRR* abs/1706.03762 (2017). arXiv: 1706.03762. URL: http://arxiv.org/abs/1706.03762.

[15] Thomas Wolf et al. *HuggingFace's Transformers: State-of-the-art Natural Language Processing*. 2020. arXiv: 1910.03771 [cs.CL].

[16] Ellery Wulczyn, Nithum Thain, and Lucas Dixon. *Ex Machina: Personal Attacks Seen at Scale*. 2017. arXiv: 1610.08914 [cs.CL].

[17] Justine Zhang et al. "Conversations Gone Awry: Detecting Early Signs of Conversational Failure". In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics, July 2018, pp. 1350–1361. DOI: 10.18653/v1/P18-1125. URL: https://www.aclweb.org/anthology/P18-1125.