Time aspect in making an actionable prediction of a conversation breakdown

Piotr Janiszewski, Mateusz Lango[0000−0003−2881−5642] (✉), and Jerzy Stefanowski[0000−0002−4949−8271]

Poznan University of Technology, Faculty of Computing and Telecommunication, Institute of Computer Science, ul. Piotrowo 2, 61-138 Poznan, Poland
1piotr.janiszewski@wp.pl, {mlango,jstefanowski}@cs.put.edu.pl

Abstract. Online harassment is an important problem of modern societies, usually mitigated by the manual work of website moderators, often supported by machine learning tools. The vast majority of previously developed methods enable only retrospective detection of online abuse, e.g., by automatic hate speech detection. Such methods fail to fully protect users as the potential harm related to the abuse has always to be inflicted. The recently proposed proactive approaches that allow detecting derailing online conversations can help the moderators to prevent conversation breakdown. However, they do not predict the time left to the breakdown, which hinders the practical possibility of prioritizing moderators' works. In this work, we propose a new method based on deep neural networks that both predict the possibility of conversation breakdown and the time left to conversation derailment. We also introduce three specialized loss functions and propose appropriate metrics. The conducted experiments demonstrate that the method, besides providing additional valuable time information, also improves on the standard breakdown classification task with respect to the current state-of-the-art method.

Keywords: online abuse· conversation breakdown prediction· time aspects in online dialog· hierarchical neural networks

1 Introduction

Cyberspace has a large potential for making constructive conversations, facilitating communication and cooperation of groups of people with similar interests, various areas of expertise. Unfortunately, some online discussions result in antisocial behaviors [16] since anonymity and an apparent sense of impunity limit the natural inhibitions interlocutors would have during a face-to-face conversation. A survey conducted in the US demonstrated that online harassment is a widespread phenomenon as approximately four-in-ten Americans were directly affected by some forms of it [8]. Online abuse can be a root cause of a wide range of mental problems, negatively affecting many aspects of victims' lives [2, 18]. Even merely witnessing the harassment on the Internet can lead to a user's lower involvement in online service or even a complete refrain from using it [27, 26].

Therefore, numerous websites leverage systems for hampering antisocial behavior. The most common methods include community moderation, up- and down-voting, the possibility to report comments, mute functionality, and banning users on the platform [5]. However, these simple approaches cannot successfully overcome the widespread problem, as a lot of hateful content can be overlooked by the moderators or simply not be reported by users. As a consequence, multiple machine learning techniques are used to support moderators by ranking unacceptable posts [9], automatically identifying cyberbulling [29] or detecting hate speech [11].

The majority of existing systems perform toxicity detection retrospectively. Even though such solutions mitigate the problem, they do not fully protect users as the potential harm has always to be inflicted to some extent, and only then the hostile comments can be filtered. These solutions do not make actionable classifications whether an online conversation is going to end in a personal attack or not, leaving no time for moderators to intervene before any harassment or conflict emerges.

A much more successful strategy would be to avert offensiveness when the discussion is still salvageable or at least hinder potential destructive effects. For instance, one could introduce to the conversation customized counter speech, which proved to be effective in combating offensiveness in various studies [23, 15]. Another solution would be to remind the interlocutors about the need for empathy and the rules of the service [17]. Even drawing moderators' attention to the derailing conversation can be beneficial as it reduces the response time and gives them an opportunity to intervene. Nevertheless, such solutions require a method for predicting conversation derailment in advance.

Moreover, just recognizing if the discussion is going to get out of hand may not be enough to obtain comprehensive and highly useful information about the potential derailment. Therefore, additional clues have to be provided. One of them is the *time to the breakdown*, which seems advantageous in many potential fields of application, especially when humans are in the loop and there is a need to prioritize actions to be performed. Such a forecast about the specific time of a breakdown may also help estimate the hostile tension in individual dialogs. This also can be a crucial hint for moderators who can recognize the most urgent cases and intervene on time. In addition, mistakes made on foreseeing how many utterances are left to the conversation derailment could be a valuable additional learning signal for the model and boost classification performance. This opens a new research and open problem since, to the best of our knowledge, such methods have not yet been proposed.

In this work, we propose a machine learning system based on deep neural networks that not only predicts whether the conservation will derail in the future but also estimates the number of utterances left to the derailment. We propose and explore three loss functions that allow for joint training of systems performing both the discussion breakdown prediction and time-to-the-breakdown estimation. We also introduce three valuable metrics for assessing the performance of models applied to foreseeing conversational breakdown with consideration of

the time aspect. An experimental evaluation shows that the proposed approach, aside from providing additional and useful information about time to the derailment for moderators, also achieves better results on the standard classification task of discussion breakdown.

2 Related works

A great deal of personal attacks in the cyberspace takes place during discussions when interlocutors disagree with each other, at least to some extent. Initially, a civil exchange may degenerate into a dispute resulting even in verbal aggression. Such "from within" derailments are potentially more dangerous and more troublesome to salvage than other types of toxicity (e.g., trolling or profanities), which a cybernaut can ignore more easily [28]. A conversation breakdown may have different faces and lead to distinct forms of antisocial behavior posing a considerable threat to the people involved.

Aside from causing emotional distress, failing conversations has also other negative impacts. For example, in online game industry, one of the main reasons leading users to stop playing the game is experiencing different kinds of toxicity during conversations with other players [26]. Therefore, it is crucial to forecast occurrences of offensiveness as a dialog develops and to make a correct prediction at the earliest possible moment, letting a moderator react appropriately. Even among Wikipedia editors community that is generally associated with well-educated people, abuse has proved to be a significant problem [27] that harms editors' willingness to further contribute.

Therefore, the problem of detecting various forms of toxicity in text data received recently considerable research attention. Methods for identifying cyberbullying [1, 29], hate speech [14, 11], doxing [24], or negative sentiment [12] proved to be useful to filter unacceptable content. Nevertheless, they focus on analyzing already posted, potentially harmful texts (so they work on historical recording).

Examining each text right before it is published creates an opportunity to identify abusive chunks on time [3, 19]. For instance, the system can ask the user to modify the toxic comment. However, asking for changing a comment or proposing its corrected version [22, 20] always requires an additional user's action, slows the exchange down, impede its natural flow and dynamics, and may discourage users from taking part in the discussion - especially when the prediction made by toxicity detector are too often incorrect. Another possibility is to remind the user about the need for empathy and rules of the service [17]; however, users who knowingly post hostile content might be completely unaffected by such a prompt. Therefore, more advanced solutions such as the introduction of customized counter-speech [15] are needed to solve the problem. Nevertheless, to apply techniques that prevent conversation failure, one needs to predict whether the conversation will derail first.

One method of foreseeing unacceptable content in online conversations was recently presented in [13]. The proposed approach determines whether any ad-

verse utterance will be published below a post on Instagram basing on the set of initial comments. Another method was presented by Zhang et al. [28] who proposed an approach for forecasting whether a conversation is going to derail basing on the initial two utterances in a discussion. The approach uses the logistic regression classifier and bag-of-words features together with specially designed problem-specific features.

The current state-of-the-art method for predicting a discussion breakdown, called Conversational Recurrent Architecture for ForecasTing (CRAFT) [4] relies on a deep neural network. The approach models a conversation flow with Hierarchical Recurrent Encoder-Decoder [25] and performs forecasting in an online fashion. All the predictions are made as a dialog develops, i.e. the prediction is updated after seeing each new utterance. Although the presented solution outperforms previous approaches, there is still some room for improvement. In particular, this approach does not take into account the moment in which a first disruptive utterance comes, ignoring the time aspect that could be very useful in practice.

3 Time aspect in prediction of conversation breakdown

In this work, we propose a new method for detecting derailing conversations that provides additional information about the *time left to the conversation breakdown*, understood as the number of utterances left to the derailment. Note that all the previously proposed methods for this task do not provide such additional information.

Being able to predict when the dialog is going to fail would bring considerable benefits in practice. For instance, the websites would be able to manage their moderation resources more effectively by prioritizing the cases of abuse, paying most attention to the most urgent and most severe ones, and counteracting them more quickly.

3.1 Proposed neural network architecture

An utterance context is a crucial factor to be considered when deciding if the utterance is abusive, as it can intensify or soften its overtone. Therefore, a breakdown should not be treated as a property of a single comment but rather as a property of a developing dialog. Following this idea, similarly to related works, the proposed method uses the hierarchical recurrent encoder-decoder (HRED) architecture [25] to model a developing dialog and to capture the conversation dynamics.

HRED consists of two recurrent neural networks called, utterance encoder and context encoder, respectively. The utterance encoder's goal is to construct a feature representation of a single user's utterance, which is then passed as an input to the context encoder. In our experiments, both networks are based on Gated Recurrent Units [6]. The input to utterance encoder is given as a sequence

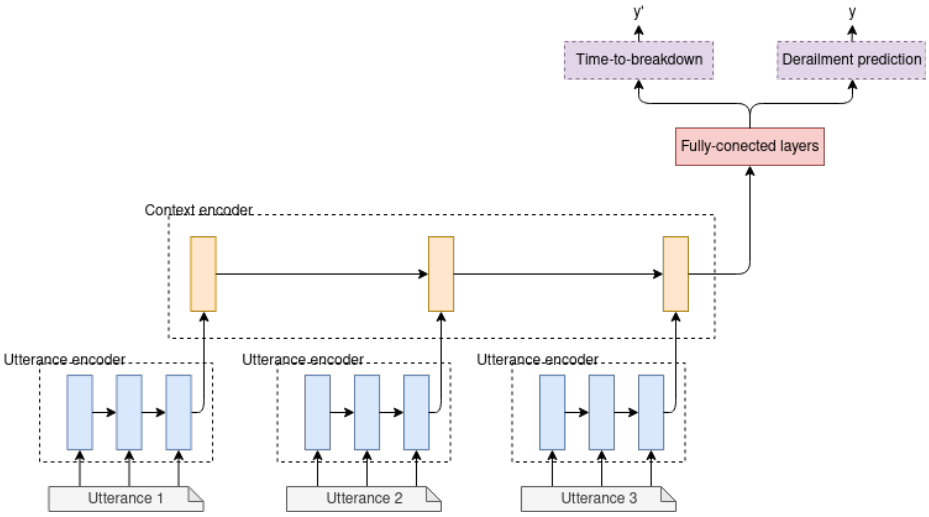

Fig. 1. The overview of the proposed neural network architecture, where y denotes the probability that the conversation will derail and y' is the prediction of the time left to the conversation breakdown.

of words, previously processed by an embedding layer. The final hidden state of the encoder is forwarded as an input to the context encoder.

In order to produce the useful feature representation for predicting both the probability of conversation failure and time-to-breakdown, the hidden state of the context encoder is passed through several fully-connected layers. Such constructed feature representation is processed by two separate output layers. The first one being the layer with only one sigmoid unit which predicts the probability that the conversation will derail in the future. The second output layer working on the same feature representation is a regression layer that predicts the time-to-breakdown (i.e., the number of utterances). The whole network architecture is trained jointly by back-propagation.

Note that when the sigmoid layer predicts that the dialog is not going to derail, the output of the regression layer can be discarded. However, the error related to the time-to-breakdown prediction provides an additional training signal to guide the model learning process. In related models without time-to-breakdown output, the error related to the derailment prediction is suffered usually only once, at the moment of conversation breakdown. Alternatively, to ensure that the model will predict possible derailment as soon as possible, one could enforce the derailment prediction after each utterance in the dialog. Nevertheless, such a solution will incorrectly introduce an association between usually the conversation beginning and the conversation breakdown class, adding unnecessary noise to the classifier training. By training the model with the additional output for time-to-breakdown we want to avoid these problems, at the same time providing a clear, additional training signal to the model.

3.2 Loss functions incorporating time aspect

In order to jointly train the network predicting the probability of conversation failure and time-to-breakdown, we propose to use the following loss function:

$$\min L(\theta) = \alpha L_{classification}(\theta) + (1-\alpha) L_{time}(\theta)$$

This loss function has two components. The first one measures the error on the standard conversation failure prediction task, while the second term controls the time prediction error. These components are weighted with the parameter $\alpha \in (0, 1)$ that controls the trade off between the model focus on the time-to-breakdown prediction and the classification task. In practice, this parameter could be tuned with the validation data, but in this work we treat both tasks as equally important, i.e $\alpha = 0.5$.

The classification error is measured by the standard cross-entropy error:

$$L_{classification}(\theta) = \frac{1}{n} \sum_{i=1}^{n} [y_i \log h_\theta(x_i) + (1-y_i) \log(1 - h_\theta(x_i))]$$

where $y_i \in \{0, 1\}$ is the label from the gold standard and $h_\theta(x_i)$ is the prediction of the classification layer.

The second term of the loss function is defined as

$$L_{time}(\theta) = \frac{1}{n} \sum_{i=1}^{n} f(g_\theta(x_i), y'_i)$$

where y'_i is the gold standard for the time-to-breakdown task, $g_\theta(x_i)$ is the prediction of the regression layer and f is a function measuring error made on a particular example. Note that the time-to-breakdown y'_i is understood as the number of utterances left to the first uncivil utterance from i-th utterance. In this work, we will explore three possible ways of measuring the time-to-breakdown error: by a classical squared error for regression, casting the task to classification, and a time-dependent custom loss.

The classical squared error is defined as:

$$f_{MSE}(g_\theta(x), y') = (g_\theta(x) - y')^2$$

which is the squared difference between predicted time $g_\theta(x)$ and the time to derailment in the gold standard y'.

Yet another possibility is to treat time-to-breakdown prediction as a classification task by defining classes for specific ranges of time-to-breakdown. We use 11 classes, where each class j corresponds to a number of utterances left to the conversation failure and $j \in \{0..10\}$. The first class i.e. $j = 0$ represents a moment of an actual derailment and class $i = 10$ means that the dialog will break down in 10 comments or more. It is assumed that a discussion horizon longer than ten utterances is so distant and so uncertain that one can aggregate these cases into a single class. Adopting such a strategy should not have any strong

negative impact on the quality of the prediction, nor on its usefulness for the potential action related to a possible failure. After casting the task to multi-class classification, we apply categorical cross-entropy error defined by

$$f_{CCE}(g_\theta(x), y') = -\sum_{j=0}^{10} y'_j \log g_\theta(x)_j$$

where $y'_j \in \{0, 1\}$ are binary variables indicating whether the time left to the breakdown belong to the class j. In this case, the activation of the respective output layer should be softmax.

We also explore the possibility of using a custom time-depended loss. The proposed loss follows the observation that the model makes more predictions for the utterances that are relatively far away from the discussion horizon in the case of long discourses. In such a case, the conversation outcome is difficult to foresee, not only because initially there can be no or little indicators that the conversation will fail but also because the prediction is based on a very small context. Therefore, in practice over- or under-estimating the time-to-breakdown about a constant value, e.g. 1, can be considered less severe if there is much time left to the derailment and considered more serious if the breakdown horizon is close.

We encompass this intuition in the following formulation:

$$f_{CTD}(g_\theta(x), y') = \begin{cases} \min\left\{\frac{|g_\theta(x)-y'|}{y'+1}; 1\right\}, & \text{for a failing conversation} \\ \max\left\{\frac{y'-g_\theta(x)+1}{y'+1}; 0\right\}, & \text{for a civil conversation} \end{cases}$$

where for the civil conversation the y' is set to be the length of the conversation. If the discussion is derailing, the presented loss is computed as the minimum of one and the absolute prediction error divided by the actual number of remaining comments. Therefore, predictions with the high horizon do not generate higher cumulative losses and the loss value is always between 0 and 1. When the discourse stays civil, the loss is equal to zero when the model anticipates that the exchange will fail even later than it actually ends; thus, one is added in the numerator. Since the loss is computed as the maximum of 0 and the forecast error divided by the true number of remaining utterances, the higher the number of comments foreseen as remaining till the conversation breakdown, the smaller the loss.

3.3 Metrics considering the time-to-breakdown of prediction

We propose three quality measures designed for the time-to-breakdown prediction. Each of them is bounded from 0 to 1 and expressed by an average of inverse errors, i.e.:

$$Q = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{E_i + 1}$$

where N is the number of dialogues in a dataset, and E_i is the prediction error of a particular dialog, defined differently for each measure. Note that when the prediction error is equal to 0, our quality measures are equal to 1. Moreover, the measure values will not be dominated by predictions on long conversations since they are averaged over conversations and not over utterances.

The first measure, denominated as *average inverse prediction errors* (AIE), uses the classic definition of absolute error, so in that measure E_i is defined as

$$E_{AIE} = \frac{1}{K} \sum_{j=1}^{K} |\hat{y}'_j - y'_j|$$

where K is the length of the conversation, $\hat{y}'_j$ is the time-to-breakdown prediction at the time j while y'_j denotes corresponding gold standard value, i.e., the true time-to-breakdown.

The second measure, called *selected inverse prediction errors* (SIE_t) focuses on the quality of prediction at the specific time point before the possible conversation breakdown, and is defined as:

$$E_{SIE_t} = \mid \widehat{y}'_t - y'_t \mid$$

where $\widehat{y}'_t$ and y'_t are the predicted and the gold standard value at t utterances *before breakdown* or before the end of conversation. We hope that this measure could be important for practitioners, assuming that in practice one should have the information about the possible conversation failure at least e.g. $t = 5$ utterances before in order to have enough time to take action.

The third measure is *inverse prediction errors* at the highest probability point (IEH):

$$E_{IEH} = \mid \widehat{y}'_{t*} - y'_{t*} \mid$$

where t^* denotes the moment in which the classifier predicted the breakdown with the highest probability. Such measure in a simplistic way takes into account that the output of the time-to-breakdown predictor will probably be used only when the classifier will assess the conversation as derailing.

Additionally, we measure the quality of time-to-breakdown prediction with standard macro-averaged F1-measure using the same eleven classes defined in the previous section for cross-entropy error. For the methods which return continuous prediction of time-to-breakdown, we round the predictions before calculating of F1-measure.

4 Experiments

The main aims of the experiments is to verify the usefulness of the new proposed approach and, in particular, to examine how the introduced loss functions influence the models' ability to predict the conversational breakdown and to approximate the time when it is going to happen. The method will be compared

against the performance of the reference method – CRAFT, which is considered as the state of the art approach. The quality of inference will be estimated by using both standard classification measures, as well as the three proposed metrics.

4.1 Datasets

In our experiments, we use the same two datasets on which CRAFT's quality has been originally measured [4].

The first dataset consists of 4188 conversations retrieved from WikiConv [10]. It contains public discussions between Wikipedia contributors about the quality of entries and observance of the Wikipedia editing rules. Crowdworkers labelled them according to whether they contain a personal attack directed towards one of the interlocutors or not. Such an act of aggression should be committed by one of the contributors who took part in the dialog since its beginning.

The second dataset contains 6842 dialogues from the subreddit ChangeMyView. A conversation is considered as derailed if it contains a comment removed by a moderator due to a violation of Rule 2: "Don't be rude or hostile to other users". It means that there may exist discussions with abusive expressions without a correct label since they could go unnoticed by the moderators. The authors of the dataset additionally warrant that every deleted comment was written by a person previously involved in the conversation.

Additionally, every example which ended with a failure is paired with a civil one on the same topic in order not to let the model associate topic-specific information with individual labels (e.g. exchanges about politics are prone to fail). Significantly, in each derailing exchange all the utterances up to the toxic one are civil.

4.2 Experimental setup

The setup of the method involves proper choosing of several architectural details in order to let the model learn effectively. In our experiments, HRED has two encoder networks (utterance and context encoder), each consisting of two GRU layers with the hidden layer of size of 500. The features for output layers are constructed by two fully-connected layers, the first one having 500 neurons and the second one with 250 units. As regularized we apply dropout with the rate of 0.1. Training batches contained 64 examples each and the process was optimized using Adam optimization algorithm with the learning rate of 10^{-5}. The end of training was determined using early-stopping in order to avoid overfitting.

Additionally, the HRED component was pre-trained on 1 million discussions from the Wikipedia Talk Page, using the generative pre-training technique proposed by the CRAFT's authors [4]. During such pre-training HRED component learns how to model the dynamics of a conversation in an unsupervised fashion.

The quality of forecasting whether a dialog will fail was measured using standard classification metrics, i.e., accuracy (Acc), precision (Prec), recall (Rec),

false positive rate (FPR), and F1-score (F1). A conversation was deemed as failing, when at least one comment was identified as derailing before the dialog failed. Each forecast was based on the previous utterances from the same conversation, thus, for first utterances nothing was predicted. The metrics were computed on the test part of datasets as provided in [4] i.e. 20% of conversations were used for testing.

When foreseeing the number of utterances left to the derailment, the metrics described in Section 3.3 were used. For the SIE_t metric, we have used $t = 5$, i.e., the error was calculated looking at the prediction triggered by the fifth to last utterance in a discussion. If the conversation was shorter than 5, the prediction on the second utterance from the beginning was taken into account.

Note that the results achieved by CRAFT reported in this work are worse than those presented in [4]. During those experiments, predictions were triggered only for the last comments in each conversation. This gave CRAFT a special advantage, as each inference was drawn basing on the complete history of the conversation, providing the model with the best possible context for its forecast. It was serious facilitation, which would not happen in real-life setting, as the horizon of a dialog is unknown, and forecasts have to be made even if the available context is too short. Moreover, such an approach also makes it impossible to measure how the model works in the complete development of the conversation.

4.3 Results of experiments

	Wikipedia Talk Pages					Reddit CMV				
Approach	Acc	Prec	Rec	FPR	F1	Acc	Prec	Rec	FPR	F1
CRAFT	0.606	0.573	0.776	0.574	0.660	0.524	0.522	0.572	0.523	0.546
MSE	**0.639**	**0.638**	0.641	**0.362**	0.640	0.546	**0.572**	0.364	**0.272**	0.445
CCE	0.616	0.597	0.710	0.479	0.649	**0.556**	0.546	**0.658**	0.547	**0.597**
CTD	0.614	0.591	**0.786**	0.554	**0.665**	0.534	0.529	0.626	0.557	0.573

Table 1. Comparison of the proposed method with the three loss functions (MSE, CCE, CTD) and the state-of-the-art CRAFT model on the task of forecasting conversational derailment.

The results of the experiments are presented in Table 1 and 2. In the classification task, one can observe that for both datasets CRAFT is outperformed by the methods proposed in this work on each of the metrics. Model which uses MSE time-to-breakdown error in the loss function achieved the best results on Wikipedia dataset, when it comes not only to accuracy, but also precision and false positive rate. These are significantly better scores compared to CRAFT. It also offered improvements on this measures on Reddit dataset, but it was CCE

loss that provided the best accuracy, recall and F1-score on that dataset. The solution based on the Custom Time Dependent loss proposed in this work improves all the classification metric with respect to CRAFT on Wikipedia data and almost all (except FPR) on Reddit dataset. This demonstrates that the information about time-to-breakdown provides a useful additional learning signal to guide model training for this conversation breakdown prediction.

	Wikipedia Talk Pages				Reddit CMV			
Approach	AIE	SIE_5	IEH	F1	AIE	SIE_5	IEH	F1
MSE	**0.480**	**0.400**	**0.572**	0.430	0.363	**0.398**	0.469	0.322
CCE	0.407	**0.400**	0.342	0.557	**0.428**	0.388	**0.686**	0.205
CTD	0.361	0.257	0.473	**0.602**	0.416	0.368	0.437	**0.370**

Table 2. Comparison between the performances of the proposed method with different loss functions on the task of predicting the number of comments left to a conversation breakdown.

In the task of approximating time-to-breakdown on Wikipedia dataset, the proposed method with MSE achieved the best results on the inverse error metrics. The result on AIE close to 0.5 means that the model is wrong on average by only one comment. In our opinion, it should be sufficient to provide an effective support for online moderators. Surprisingly, our Custom Time Dependent Loss and not the standard cross-entropy provided better results on F1-score, i.e., while evaluating time-to-breakdown prediction as a multi-class classification task. On the Reddit dataset, CTD also gave the highest F1-score, but it was CCE that gave the highest values of AIE and IEH measures.

Note that the values of SIE_5 are generally lower than values of AIE and IEH for both datasets. This is because the prediction error taken into account when calculating SIE is calculated 5 comments before a personal attack, and the dialog context is often not sufficiently broad to make a good prediction. Nonetheless, this metric allows to check, what is the forecast quality, when the conversation is not completely developed and there is still much time to intervene. According to the definition of SIE the average number of conversations for which the best model was wrong is 1.5, which is a satisfactory result considering how early this prediction is made.

Furthermore, IEH values are usually higher than AIE and SIE_5 for most of approaches. This is due to the fact that the probability of derailment increases as the conversation develops in the failing direction and subsequent forecasts are made basing on wider contexts. This implies that as the model becomes more and more convinced that the exchange will eventually fail, it can more accurately foresee when it is going to happen. This is a good characteristics, as in case of dialogs with high tension (thus easier to detect) the final conflict should be potentially more serious, therefore it is especially important to identify how many

comments are left to such conversation breakdown. For our best solution, the committed average error is only around 0.75 and 0.45 utterances for respective datasets.

5 Conclusions

This work introduces a new version of the online abuse conversation breakdown problem, which includes jointly predicting whether the conversation will derail and approximating time to the conversation breakdown. In particular, considering time aspects opens new research and application perspectives. Upon the current state-of-the-art, we presented a new approach to this problem by proposing three task-specific loss functions and extending the hierarchical recurrent neural network architecture.

The experiments with two datasets containing different real life online discussions have showed that the proposed methods (with these loss functions) achieve better results on the accuracy, F1-score, precision, and recall measures than the current state-of-the-art method for conversation breakdown prediction. Additionally, the proposed approach returns new type of information about time-to-breakdown, which could be very helpful in practice, for instance, to prioritize the cases handled by moderators.

Nevertheless, the approach described in this work could be still further developed. One possible option is to use a pre-trained architecture that models conversation dynamics in another way than HRED. In particular, recent experiments with the transformer-based models in many related natural language processing tasks may suggest that using the neural networks of this type may boost the results. Therefore, we also carried out some experiments by using contextual word embeddings produced by one light-weight transformer-based model, namely DistilBERT [21], but the results were not clear enough. Most importantly, the use of DistilBERT embeddings never produced better results than those obtained by any of the new proposed HRED-based methods, even though we have seen some improvements for some particular configurations of dataset and loss function. Nevertheless, we hypothesize that proposing a new transformer-based architecture dedicated to modeling conversations could be a topic of further research.

The other possible issue is that our model, similarly to related works, has been trained on a balanced dataset, even though online conversation derailments happen relatively less frequently. Therefore, while the system should be able to deal with a shifted class distribution. The question of how the low number of positive examples may influence the predictive performance of conversation breakdown predictors is still open.

Finally, the more advanced ways of dealing with the time aspect in predicting a conversation breakdown can be further explored. For instance, one can try to adopt early classification methods [7] that, instead of predicting time-to-breakdown, are directly trying to optimize the trade-off between the quality and earliness of event prediction, which can be useful in practice. Another possibility

would be to explore ideas from the field of survival analysis or from the next event prediction problem in time series.

Acknowledgements

Mateusz Lango was supported by the Polish National Science Centre under grant No. 2016/22/E/ST6/00299. Moreover, the research of Jerzy Stefanowski was partially supported by the Polish Ministry of Education and Science, grant no. 0311/SBAD/0709. The authors also acknowledge the support from Google Cloud Platform research grant.

References

1. Agrawal, S., Awekar, A.: Deep learning for detecting cyberbullying across multiple social media platforms. In: Pasi, G., Piwowarski, B., Azzopardi, L., Hanbury, A. (eds.) Advances in Information Retrieval. pp. 141–153. Springer International Publishing, Cham (2018)
2. Beran, T., Li, Q.: Cyber-harassment: A study of a new method for an old behavior. Journal of educational computing research **32**(3), 265 (2005)
3. Carton, S., Mei, Q., Resnick, P.: Extractive adversarial networks: High-recall explanations for identifying personal attacks in social media posts. Proceedings of the Conference on Empirical Methods in Natural Language Processing (2018)
4. Chang, J.P., Danescu-Niculescu-Mizil, C.: Trouble on the horizon: Forecasting the derailment of online conversations as they develop. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (2019)
5. Cheng, J., Danescu-Niculescu-Mizil, C., Leskovec, J.: Antisocial behavior in online discussion communities. In: Proceedings of ICWSM (2015)
6. Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using RNN encoder-decoder for statistical machine translation. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (2014)
7. Dachraoui, A., Bondu, A., Cornuéjols, A.: Early classification of time series as a non myopic sequential decision making problem. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases (pp. 433-447). Springer (2015)
8. Duggan, M.: Online harassment 2017, Pew Research Center (2017)
9. Hsu, C.F., Khabiri, E., Caverlee, J.: Ranking comments on the social web. In: 2009 International Conference on Computational Science and Engineering. vol. 4, pp. 90–97. IEEE (2009)
10. Hua, Y., Danescu-Niculescu-Mizil, C., Taraborelli, D., Thain, N., Sorensen, J., Dixon, L.: WikiConv: A corpus of the complete conversational history of a large online collaborative community. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. pp. 2818–2823. Association for Computational Linguistics, Brussels, Belgium (2018).
11. Janiszewski, P., Skiba, M., Walińska, U.: Pum at semeval-2020 task 12: Aggregation of transformer-based models' features for offensive language recognition. In: Proceedings of the International Workshop on Semantic Evaluation (SemEval) (2020)

12. Lango, M.: Tackling the problem of class imbalance in multi-class sentiment classification: An experimental study. Foundations of Computing and Decision Sciences **44** (2019)
13. Liu, P., Guberman, J., Hemphill, L., Culotta, A.: Forecasting the presence and intensity of hostility on instagram using linguistic and social features. Proceedings of AAAI Conference on Web and Social Media (ICWSM) (2018)
14. Malmasi, S., Zampieri, M.: Detecting hate speech in social media. Proceedings of the International Conference Recent Advances in Natural Language Processing (2017)
15. Mathew, B., Saha, P., Tharad, H., Rajgaria, S., Singhania, P., Maity, S.K., Goyal, P., Mukherjee, A.: Thou shalt not hate: Countering online hate speech. In: Proceedings of the International AAAI Conference on Web and Social Media. vol. 13, pp. 369–380 (2019)
16. Mishra, P., Yannakoudakis, H., Shutova, E.: Tackling online abuse: A survey of automated abuse detection methods, CoRR (2019)
17. Munger, K.: Tweetment effects on the tweeted: Experimentally reducing racist harassment. Political Behavior **39**(3), 629–649 (2017)
18. Munro, E.R.: The protection of children online: a brief scoping review to identify vulnerable groups. Childhood Wellbeing Research Centre (2011)
19. Noever, D.: Machine learning suites for online toxicity detection. arXiv preprint arXiv:1810.01869 (2018)
20. Prabhumoye, S., Tsvetkov, Y., Salakhutdinov, R., Black, A.W.: Style transfer through back-translation. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (2018)
21. Sanh, V., Debut, L., Chaumond, J., Wolf, T.: Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. Proceedings of the 5th Workshop on Energy Efficient Machine Learning and Cognitive Computing (co-located with NeurIPS) (2019)
22. Santos, C.N.d., Melnyk, I., Padhi, I.: Fighting offensive language on social media with unsupervised text style transfer. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (2018)
23. Schieb, C., Preuss, M.: Governing hate speech by means of counterspeech on facebook. In: 66th ICA Annual Conference. pp. 1–23 (2016)
24. Snyder, P., Doerfler, P., Kanich, C., McCoy, D.: Fifteen minutes of unwanted fame: Detecting and characterizing doxing. In: Proceedings of the 2017 Internet Measurement Conference. pp. 432–444 (2017)
25. Sordoni, A., Bengio, Y., Vahabi, H., Lioma, C., Simonsen, J.G., Nie, J.Y.: A hierarchical recurrent encoder-decoder for generative context-aware query suggestion. In: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management (2015)
26. Stoop, W., Kunneman, F., van den Bosch, A., Miller, B.: Detecting harassment in real-time as conversations develop. In: Proceedings of the Third Workshop on Abusive Language Online. pp. 19–24. Association for Computational Linguistics, Florence, Italy (Aug 2019). https://doi.org/10.18653/v1/W19-3503
27. Wulczyn, E., Taraborelli, D., Thain, N., Dixon, L.: Ex Machina: Personal Attacks Seen at Scale, International World Wide Web Conferenc (2017)
28. Zhang, J., Chang, J.P., Danescu-Niculescu-Mizil, C., Lucas Dixon, Y.H., Thain, N., Taraborelli, D.: Conversations gone awry: Detecting early signs of conversational failure. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics (2018)

29. Zhao, R., Zhou, A., Mao, K.: Automatic detection of cyberbullying on social networks based on bullying features. In: Proceedings of the 17th International Conference on Distributed Computing and Networking. ICDCN '16, Association for Computing Machinery, New York, NY, USA (2016). https://doi.org/10.1145/2833312.2849567