Multimodal Conversation Modelling for Topic Derailment Detection

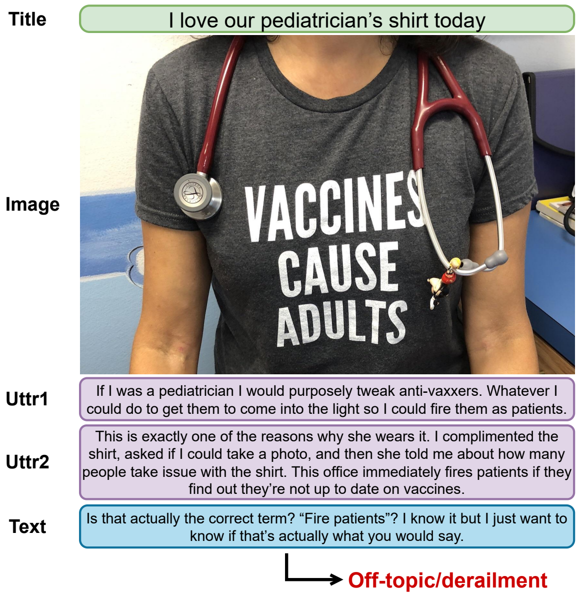
Zhenhao Li, Marek Rei, Lucia Specia
Language and Multimodal AI (LAMA) Lab, Imperial College London
{zhenhao.li18, marek.rei, l.specia}@imperial.ac.uk

Abstract

Conversations on social media tend to go off-topic and turn into different and sometimes toxic exchanges. Previous work focuses on analysing textual dialogues that have derailed into toxic content, but the range of derailment types is much broader, including spam or bot content, tangential comments, etc. In addition, existing work disregards conversations that involve visual information (i.e. images or videos), which are prevalent on most platforms. In this paper, we take a broader view of conversation derailment and propose a new challenge: detecting derailment based on the "change of conversation topic", where the topic is defined by an initial post containing both a text and an image. For that, we (i) create the first Multimodal Conversation Derailment (MCD) dataset,[1] and (ii) introduce a new multimodal conversational architecture (MMConv) that utilises visual and conversational contexts to classify comments for derailment. Experiments show that MMConv substantially outperforms previous text-based approaches to detect conversation derailment, as well as general multimodal classifiers. MMConv is also more robust to textual noise, since it relies on richer contextual information.

1 Introduction

Online conversations on social media can easily go off-topic (Churchill and Bly, 2000; Shepherd et al., 2015) and result in divergent and even harmful exchanges, ranging from off-topic discussions to spam information and personal attacks. Work in this field has focused on detecting or forecasting conversations that derail into toxic content (Zhang et al., 2018a,b; Chang and Danescu-Niculescu-Mizil, 2019; Kementchedjhieva and Søgaard, 2021; Lambert et al., 2022). However, such work is limited in two ways: 1) derailment is not limited to toxic comments or personal attacks; it can include topic shifts, spam posts, among others; 2) conversations in real applications can include modalities other than text, such as image, audio, and video, and the multimodal context can be crucial in deciding whether a conversation has derailed.



I love our pediatrician's shirt today

If I was a pediatrician I would purposely tweak anti-vaxxers. Whatever I could do to get them to come into the light so I could fire them as patients.

This is exactly one of the reasons why she wears it. I complimented the shirt, asked if I could take a photo, and then she told me about how many people take issue with the shirt. This office immediately fires patients if they find out they're not up to date on vaccines.

Is that actually the correct term? "Fire patients"? I know it but I just want to know if that's actually what you would say.

Figure 1: An example of multimodal conversation derailment detection. The conversation starts with a title and an image, with a sequence of utterances replying to each other and a target text which is classified to be a derailment or non-derailment.

In this work, we focus on a broader view on detecting conversation derailment based on the change of conversation topic, making it a more challenging task, and propose the task of multimodal derailment detection. Figure 1 shows an example of multimodal conversation derailment detection which includes multiple input types. We collect data from Reddit and use crowd-annotation to locate derailments, creating a multimodal Conversation Derailment (MCD) dataset. To the best of our knowledge, this is the first work introducing multimodality in conversation derailment detection. In addition, we propose a new multimodal conver-

[1] https://github.com/Nickeilf/Multimodal-Conversation-Derailment

Findings of the Association for Computational Linguistics: EMNLP 2022, pages 5115–5127
December 7-11, 2022 ©2022 Association for Computational Linguistics

sational (MMConv) architecture which makes use of image and conversational contexts via a hierarchical structure to improve the performance of derailment detection. To simulate real situations on social platforms, we also perform a robustness test by evaluating the model under noisy texts. To summarise, our main contributions are:

- We propose a new task: multimodal conversation derailment detection, and create a new dataset for this challenging task.
- We propose a new hierarchical model architecture – MMConv – that uses visual and conversational contexts. Experiments show that our model outperforms existing multimodal architectures in this task.
- We evaluate the model robustness against noisy texts, showing that images and conversations can provide useful context and improve model robustness.

2 Related Work

Conversation derailment detection Previous work in conversation derailment detection focuses on derailment due to antisocial behaviours. Zhang et al. (2018a) propose the Conversations Gone Awry dataset and introduce the task of forecasting antisocial behaviours in conversations. Chang and Danescu-Niculescu-Mizil (2019) propose the Reddit CMV dataset and a CRAFT model that can potentially provide early warnings regarding abusive content. Kementchedjhieva and Søgaard (2021) further introduce dynamic training into BERT-based (Devlin et al., 2019) derailment forecasting models. Other studies investigate general toxicity in conversations (Almerekhi et al., 2020; Cécillon et al., 2020; Ive et al., 2021; Vidgen et al., 2021; Lambert et al., 2022) regardless of conversation derailment. In this work, we study derailment detection from a broader angle, based on the change of conversation topics. The work by Park et al. (2021) is the closest to ours. They study conversations violating a set of community norms, such as uncivil behaviour, off-topic content, etc. However, their work only considers textual conversations, whereas we look at multimodal conversations.

Conversation modelling Existing approaches to modelling conversations can be divided into two categories: hierarchical modelling and concatenated modelling. In the first category, each utterance in the conversation is encoded first into an utterance vector, and these utterance vectors are further encoded to generate a conversation vector. Sordoni et al. (2015) propose the HRED architecture, which uses two RNNs as the utterance encoder and context encoder. The utterance and context encoder have also been modelled with GRUs (Yang et al., 2016), LSTMs (Chang et al., 2019), and BERT (Zhang et al., 2019; Pappagari et al., 2019; Li et al., 2020; Gu et al., 2021; Santra et al., 2021). The second category treats the conversation as a concatenation of all utterances, separated by special tokens, and feeds it to Transformer-based (Vaswani et al., 2017) pre-trained language models to generate a vector representing the whole conversation (Lai et al., 2020; Zhong et al., 2021). This approach can be more effective, but requires more computational resources to handle the long inputs. We adopt the first type of approach to model the conversations.

3 MCD Task and Dataset

3.1 Task definition

On popular social platforms such as Reddit, a conversation often starts with an original post, which consists of a segment of text, sometimes with images or a paragraph of text. Users can then either comment under the post or reply to the comments. This means the post has a tree structure, where each branch can be seen as a conversation. Following this natural feature of a conversation, we define the multimodal dialogue derailment task, which has four types of inputs: a title T, an image I, a sequence of previous utterances (shortened to "utterances" in later paragraphs) $U = [U_1, U_2, ..., U_n]$ where n is the length of the sequence of utterances. The title and the image are posted by the same user and are complementary to convey the topic. Each utterance U_i is the reply to the previous utterance U_{i-1}, and the target text X is the reply to the last utterance U_n. The utterances sequence can be empty, and in this case the target text is the direct reply to the title/image. This task aims to detect whether the target text has derailed the conversation. The target text is considered as a derailment if:

1) The target text does not directly relate to the initial topic. The initial topic can include the subject of the image, the image itself (its quality, colour, lighting, etc.), and the title itself (the way it is formulated).

2) The target text refers to the content of the

image or title, but starts a different conversation topic on them.

3) The target text is spam (e.g. advertising, unintelligible, etc.).

4) The target text is toxic (e.g. personal attack, insulting, etc.).

Figure 2 shows examples of the four types of derailments.

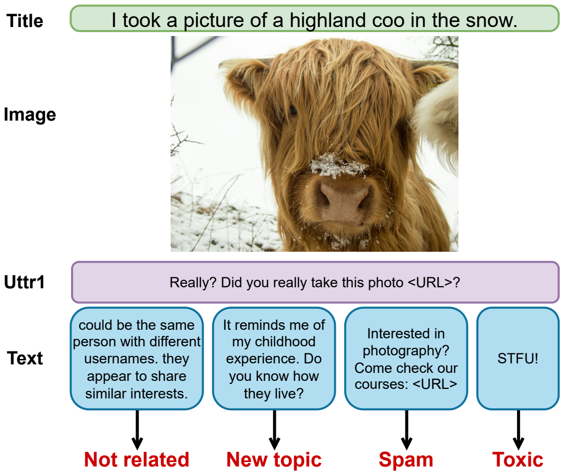

Title
I took a picture of a highland coo in the snow.
Image
Uttr1
Really? Did you really take this photo <URL>?
Text
could be the same person with different usernames. they appear to share similar interests.
It reminds me of my childhood experience. Do you know how they live?
Interested in photography? Come check our courses: <URL>
STFU!
Not related
New topic
Spam
Toxic

Figure 2: Example of different types of derailments.

3.2 Data collection

Our data is sourced from Reddit using the Pushshift API[2] (Baumgartner et al., 2020). The data is taken from two subreddits with abundant images: /r/itookapicture and /r/pics. To select conversations with potential off-topic target texts, we focus on conversations based on three criteria:

1) Target text with keywords "off-topic", "off topic".[3]

2) Target text removed by moderators. We first collect target texts with "[removed]" as the content using Reddit API[4], and retrieve the texts before removal using Pushshift.

3) Target text with a Reddit score below -10. The score is calculated with the number of upvotes minus the number of downvotes.

For potential negative examples (on-topic target text), we select the target text with a Reddit score above 10. After collecting the target texts, we use the Reddit API to iteratively fetch the previous utterances in the reply chain, until the title and the image.

3.3 Annotation procedure

The data is annotated by qualified US-based annotators from Amazon Mechanical Turk (MTurk). The worker qualification process can be found in Appendix A. For each conversation, the annotator is asked to label whether the target text is a derailment with respect to the initial topic of the conversation, which is determined by the title and the image. The possible labels include "derailment", "non-derailment", and "skip (hard to decide)". If a derailment is chosen, the annotator is asked to also select the reason why the target text is a derailment. The possible reason labels are: "unrelated to title", "unrelated to image", "unrelated to title and image", "new conversation topic", "spam comment", and "toxic comment".

For quality control, we also include gold data with gold labels in the annotation process. The annotators' performance on this gold data is monitored during the annotation. If an annotator skips more than 10% conversations or has an accuracy below 60% on the gold data, this annotator will be blocked from further annotations. Each conversation is annotated by three annotators. The annotation interface is shown in Appendix B.

After the annotation is done, we filter the data by removing the examples annotated by blocked workers and examples with "skip" as the majority label. The average annotation accuracy on the gold data is 82%, and the Fleiss Kappa is 0.368. This Kappa score reflects a *fair agreement*. This is because the task of derailment detection can be very subjective. Therefore, we use the majority labels from the three annotators as the final label.

3.4 Statistics

The final MCD dataset includes 12,653 conversations with 5,269 images. Figure 3 shows the conversation length distribution, where the length includes the number of context utterances plus the target text. When the conversation is short, non-derailment (NON) accounts for 60% of the examples. When the conversation length is longer than three, derailment becomes the main label. This behaviour is expected – the likelihood of a target text going off-topic increases as the conversation expands into more utterances. The longest conversation chain in the MCD dataset contains 56 messages.

We randomly split the dataset into train/dev/test sets with a ratio of 0.8/0.1/0.1 (Table 1).

[2] https://github.com/pushshift/api

[3] We remove the sentences containing these keywords in the text to avoid potentially biasing the annotators.

[4] https://github.com/praw-dev/praw

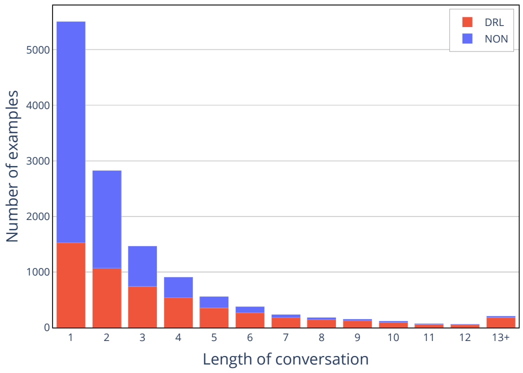


Figure 3: The conversation length distribution of MCD dataset. DRL: derailment. NON: non-derailment.

	DRL	NON	Avg Length
Train	4,250	5,872	2.74
Dev	531	734	2.88
Test	532	734	2.82

Table 1: Statistics of the train/dev/test splits. DRL: derailment. NON: non-derailment.

4 Model

In this section, we introduce the architecture of our proposed MMConv model, which consists of a pre-trained text encoder, a visual encoder, a context encoder, and a late fusion component. Figure 4 gives an overview of the model architecture.

Pre-trained text encoder We use a pre-trained BERT (Devlin et al., 2019) model as our text encoder, but note that any other pre-trained language model can be used. Given an input text sequence $\boldsymbol{\omega} = [\omega^1, \omega^2, ..., \omega^k]$, BERT encodes the input into a sequence of vectors $\mathbf{e} = [e^1, e^2, ..., e^k]$. For each utterance U_i and the target text X, we use the first hidden vector (i.e. the `[CLS]` token representation) as the sentence representation $s_{u_j} = e^1_{u_j}$ where $j \in \{1, 2, ..., n\}$ and $s_x = e^1_x$.

For the title, the input is provided as "`[CLS]` Title `[SEP]` Text `[SEP]`". Instead of using the `[CLS]` representation, we keep the hidden representations of all tokens in the title input, thus yielding $\mathbf{e}_t$.

Visual encoder We use an image object detector as our visual encoder. Given an input image I, the visual encoder detects local regions in the image as object proposals and extract m pooled visual features $\mathbf{O} = [o^1, o^2, ..., o^m]$ for each object region proposal. An MLP layer with gelu activation function (Hendrycks and Gimpel, 2016) is used to map the image features into the same space as the text, thus generating the visual hidden vectors $\mathbf{V} = [v^1, v^2, ..., v^m]$:

$$\begin{aligned} o^i &= \mathbf{Enc}_v(I), \\ v^i &= \text{gelu}(W_v o^i + b_v), \\ \forall i &\in \{1, 2, ..., m\} \end{aligned} \tag{1}$$

Given that both title and image form the initial topic of a conversation, we concatenate the title token vectors $\mathbf{e}_t$ with the visual hidden vectors $\mathbf{V}$ and fuse them using a self-attention layer (Vaswani et al., 2017) to generate a multimodal title-image hidden vector h_{mm} (the title `[CLS]` position is used for the hidden vector):

$$h_{mm} = \mathbf{SA}([e^1_t, e^2_t, ..., e^k_t, v^1, v^2, ..., v^m]) \tag{2}$$

where k is the length of the title input.

Context encoder To better model the chronological order in the sequence of utterances, we use a 1-layer uni-directional GRU network (Cho et al., 2014) as our context encoder. Given the sentence representations for each utterance $[s_{u_1}, s_{u_2}, ..., s_{u_n}]$, the context encoder iteratively computes the context hidden state h_{u_i} up to the i-th utterance:

$$h_{u_i} = \mathbf{GRU}(h_{u_{i-1}}, s_{u_i}) \tag{3}$$

where $i \in \{1, 2, ..., n\}$. Therefore, the last hidden state h_{u_n} can be viewed as the encoding of the whole sequence of utterances.

Late fusion Finally, a self-attention layer is used to fuse the multimodal title-image hidden vector h_{mm}, the utterances sequence hidden vector h_{u_n}, and the target text representation s_x, generating a vector h_{conv} encoding the whole conversation. The three hidden vectors are averaged after the self-attention layer, then fed to an MLP classifier to make the final prediction:

$$\begin{aligned} h_{conv} &= \text{avg}(\mathbf{SA}([h_{mm}; h_{u_n}; s_x])) \\ p &= \text{sigmoid}(W_c h_{conv} + b_c) \end{aligned} \tag{4}$$

5 Experiments

5.1 Baselines

We compare to a number of text-only, conversational and multimodal baselines:

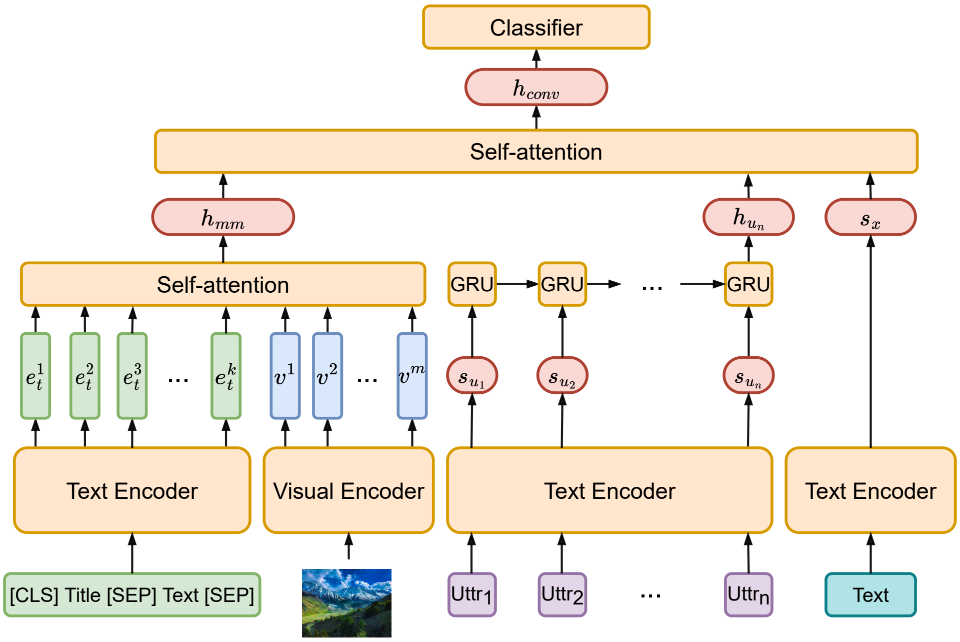


Figure 4: MMConv architecture.

WordOverlap (T, X)[5] This is a rule-based approach which simply defines a derailment when there are no overlapping contents between the title T and the text X.

BERT-simple (T, X) **(Devlin et al., 2019)** This uses a pre-trained BERT for classification. The title T and the text X are encoded using BERT separately, and their `[CLS]` token representations are concatenated and fed to a classifier head.

BERT-concat (T, U, X) This approach makes use of the sequence of utterances by concatenating the title T, utterances U, and the text X as a sequence, which is then given as input to BERT. The sequences are separated by the `[SEP]` token. The previous utterances are truncated when the concatenated sequence exceeds the 512 tokens.

CRAFT (T, U, X) **(Chang and Danescu-Niculescu-Mizil, 2019)** The CRAFT model employs the HRED architecture for modelling conversations. The CRAFT model is first pre-trained on the contextual dialog generation task using Reddit data. When fine-tuned on the derailment task, the decoder is removed and a classifier is attached to make the prediction.

ContextRNN (T, U, X) **(Ive et al., 2021)** This approach first encodes the title T, each utterance U_i in the conversation history, and the target text X separately using BERT. The `[CLS]` representations of the utterances are summarized into a context vector using a GRU network. Finally, the context vector, title vector, and the text vector are then concatenated to make the final prediction.

ToBERT (T, U, X) **(Pappagari et al., 2019)** This approach first encodes title T, utterances U, and the text X separately, but the context vector is modelled with an additional Transformer encoder block (Vaswani et al., 2017) with a positional embedding, taking the hidden representations of the utterances as input.

BERT-ViT (T, I, X) **(Dosovitskiy et al., 2021)** This approach encodes the concatenation of title T and text X (separated by `[SEP]` token) with BERT to get the `[CLS]` representations. The image I is encoded with the ViT model and is represented as a pooled visual feature vector. The joint title-text vector and the visual vector are concatenated to make the final prediction.

MMBT (T, I, X) **(Kiela et al., 2019)** Given that the title and image constitute the initial topic of the conversation, this approach early fuses title T and image I using BERT. The visual feature is extracted using ResNet-152 (He et al., 2016) and is concatenated with the joint title-text embedding (separated by `[SEP]` token) before the BERT layers.

[5] The symbols denote which types of inputs the model receives

ViLT (T, I, X) **(Kim et al., 2021)** The ViLT model is pre-trained on multiple vision-and-language tasks for better multimodal representation. The title T and the image I are encoded using the pre-trained ViLT model to get a fused vector, which is then concatenated with the text vector extracted from BERT for the prediction.[6]

5.2 Pre-processing and training

We lowercase all texts, replacing emojis and emoticons with corresponding word phrases (e.g. "=D" is replaced with "<happy>") using the *ekphrasis* toolkit (Baziotis et al., 2017). Due to computational constraints, we only use the first and last 5 previous utterances if the length of utterances sequence is above 10. We extract the "bottom-up-top-down" (BUTD) (Anderson et al., 2018) object features for the input images with a pre-trained Faster R-CNN ResNet-101 (Ren et al., 2015) object detector. The visual features are represented with 36 object vectors $\mathbf{O} \in \mathbb{R}^{36\times 2048}$.

We use the pre-trained BERT model (Devlin et al., 2019) (`bert-base-uncased`) as our text encoder. The model hidden size is 768, and the dropout rate (Srivastava et al., 2014) is set to 0.2. We use binary cross-entropy loss as our training loss and Adam (Kingma and Ba, 2015) as optimiser with an initial learning rate of 3e-5. The training batch size is 64. We train the model for 15 epochs, with early stopping when the model does not improve on the development set for 5 epochs. The best checkpoint on the development set with highest macro-F1 score is selected for evaluation on the test set. All models are trained with 5 random seeds and the average results are reported. The experiments are done on an RTX A6000 GPU.

6 Results

6.1 Benchmark

We first compare the performance of the baselines and our MMConv model on the test split. The results are shown in Table 2, where we report the accuracy, precision, recall, F1, macro-weighted F1 (F1-mw), and the area-under-curve (AUC) score.

The results for text-only models (taking only title/text as inputs) are presented in the second and third rows.The rule-based baseline WordOverlap only achieves a performance of 0.34 F1 score, which is even lower than random guessing (0.44 F1), suggesting that the task of detecting derailment is challenging and more advanced methods are needed. The BERT-simple beseline outperforms the rule-based methods by a large margin.

Comparing the conversational models (taking title/utterances/text as inputs) with the text-only models, we find that the context utterances needs to be handled separately. By simply concatenating all textual inputs into a whole sequence, the BERT-concat model shows a slight performance drop when utilising the previous utterances, compared to the BERT-simple baseline. The drop might be caused by the long sequence length resulting from concatenation, thus the model might have fewer attention weights for title and text tokens as there are too many utterance tokes for attention. Models that encode each utterance separately – CRAFT, ContextRNN, and ToBERT – all improve over the BERT-simple baseline, which shows that utilizing previous context is beneficial for this task.

It is also noticeable that CRAFT, although much smaller in model size, performs on par with BERT-based models. Gururangan et al. (2020) show that pre-training on in-domain data and a related task could result in significant benefits, therefore we hypothesize that the improvement is from the pre-training on Reddit data (same domain) and the dialogue generation task (contextual task). To test that, we train CRAFT from scratch without loading the pre-trained checkpoint from Chang and Danescu-Niculescu-Mizil (2019). The performance drops by 0.03 F1 score compared to the pre-trained CRAFT, which confirms the benefits of in-domain pre-training.

Regarding the multimodal models (taking title/image/text as inputs), both BERT-ViT and MMBT show better performance than the BERT-simple baseline from using the visual information. However, the performance of ViLT is similar to that of BERT-simple, with a lower F1 score, but higher F1-mw score. Limited by the max length of 40 tokens for text inputs, ViLT seems to lose information when encoding the title, thus causing the performance to be worse than other multimodal counterparts.

Finally, MMConv, which takes all inputs, outperforms all other approaches, improving over BERT-simple by around 0.03 F1. Inspired by Bouritsas et al. (2018), we also train a version of MMConv by replacing the visual object features with the se-

[6] Since the pre-trained ViLT model only encodes text with less than 40 tokens, we do not concatenate the title with text using the `[SEP]` token.

Models	visual	conv	Acc	Pre	Rec	F1	F1-mw	AUC
Random			.484	.402	.478	.437	.487	N/A
WordOverlap			.455	.346	.338	.342	.452	N/A
BERT-simple			.731	.668	.709	.688	.731	.810
BERT-concat		✓	.727	.673	.679	.676	.727	.803
CRAFT		✓	.744	**.705**	.674	.689	.743	.812
CRAFT w/o pre-train		✓	.714	.661	.657	.659	.714	.788
ContextRNN		✓	.747	.685	**.730**	.707	.747	.827
ToBERT		✓	.740	.683	.709	.696	.741	.820
BERT-ViT	✓		.750	**.705**	.697	.701	.750	.837
MMBT	✓		.749	**.705**	.689	.697	.748	.824
ViLT	✓		.735	.684	.676	.680	.733	.817
MMConv	✓	✓	**.756**	.704	.724	**.714**	**.756**	.840
MMConv-VC	✓	✓	.753	.699	.721	.710	.753	**.841**

Table 2: Comparison of baselines and MMConv on the test split. *-VC: model using visual concepts instead of visual features. visual: whether the model uses images as inputs. conv: whether the model uses previous utterances as inputs.

quence of object labels in a textual format, encoding it with BERT as the visual encoding. The model trained with visual concepts shows a similar performance as the one using object features, further validating the benefit from using visual information. By using the context utterances and the image as additional inputs, MMConv obtains higher precision, recall, and AUC scores than BERT-simple, which suggests that the improvement is not caused by trade-off between precision and recall.

6.2 Ablation study

To understand which components contribute the most to MMConv, we perform an ablation study by changing one model component at a time (Table 3). We first change the self-attention fusion, where the hidden vectors of title/image, context, and main text are fused, with a concatenation of the three vectors (w/o SA fusion). This change results in a small drop in F1 while also decreasing the recall. Removing the context/visual encoder (w/o context enc and w/o visual enc) causes a larger performance drop in F1, confirming the benefit of using image and previous utterances as additional context. In addition, using both context and visual encoders results in much higher recall than only using one. This indicates that the two sources of information are complementary and both provide useful information. Finally, to further analyse the effect of visual information, we perform an incongruent inference (Elliott, 2018), where we give as input visual features from a randomly selected image. Using incongruent visual features decreases both precision and recall, validating that the model indeed uses information from the provided image.

Models	Pre	Rec	F1	F1-mw
MMConv	.704	**.724**	**.714**	**.756**
w/o SA fusion	.704	.716	.710	.755
w/o context enc	.700	.690	.695	.746
w/o visual enc	**.725**	.662	.692	.751
w/ incongruent visual	.689	.720	.704	.747

Table 3: Ablation study when only one model component is changed each time.

6.3 Forecasting future derailment

Zhang et al. (2018a) and Chang and Danescu-Niculescu-Mizil (2019) propose forecasting conversation derailment as a more challenging task for detecting online dialogue derailment. The task is to test if the model can predict a derailment given only historical conversation utterances. In other words, the model has no access to the actual target text, and has to predict derailment based on the sequence of conversation utterances. Following their settings, we evaluate the conversational models on the derailment examples in the test set and calculate the mean-H (average forecast horizon), which represents how many utterances ahead the model can signal a future derailment (Table 4). All the four conversational models, although not trained specifically for forecasting derailment, can achieve a mean-H above 3 utterances, i.e. the models are capable of early warning potential derailments. Taking images as additional inputs, MMConv exhibits the largest mean-H (the mean-H distribution for MMConv model is shown in Appendix D). CRAFT outperforms ContextRNN and ToBERT in forecast-

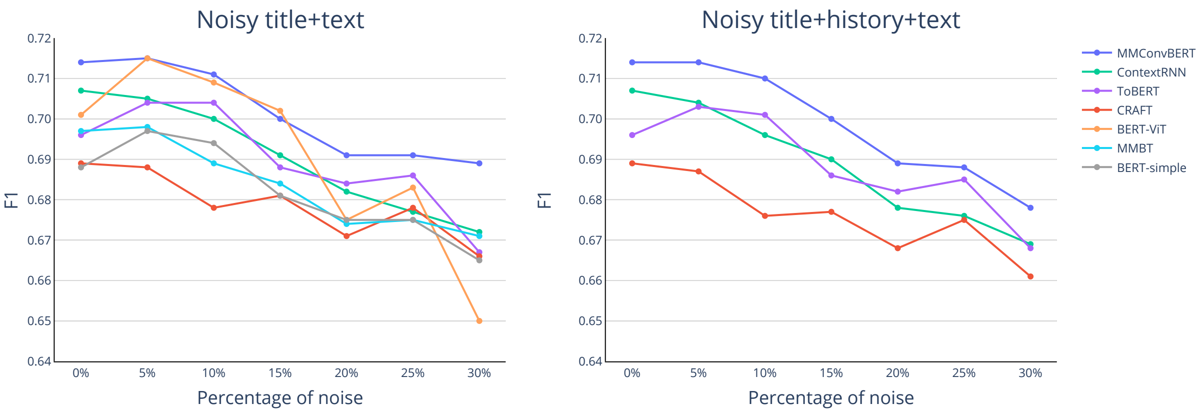


Figure 5: Robustness analysis on noisy test set. The model performance is evaluated on noisy title+text, and noisy title+history+text settings.

ing, most likely due to its utterance-level dialogue generation pre-training, which makes the model suitable for predicting future contexts.

Models	Mean-H
CRAFT	3.391
ContextRNN	3.057
ToBERT	3.247
MMConv	**3.529**

Table 4: Results of forecasting derailment for conversational models. The mean-H indicates how many utterances ahead the model can signal a derailment.

7 Robustness Analysis

Since the main application of dialogue derailment detection is automatic moderation on social platforms, where the user-generated texts can be noisy, it is important to evaluate the model's robustness to textual noise, e.g. typos, code-switching, etc. Li et al. (2021) evaluate on test data injected with noise and show that multimodal models are more robust to textual noise than unimodal models. Following their settings, we select 7 types of common textual noise (details in Appendix E) from NL-Augmenter (Dhole et al., 2021) and apply noise injection to the test set. We evaluate under two settings: noise injected to both title and target text, and a more realistic setting where noise is injected to all textual inputs, including title, previous utterances, and the target text. The models' F1 scores on different proportion of noise are shown in Figure 5.

When noise is only injected to title and text, all models show robustness against a small proportion (less than 5%). However, when the noise percentage reaches 15%, all models exhibit larger performance drops while the performance of MMConv is more stable and clearly shows less performance drop over the others, which indicates that the benefits from using image and context are consistent and can help robustness.

When noise is injected to all textual inputs, similar performance drops is observed for all conversational models. Comparing the performance with and without noise injected to previous utterances, we notice that the difference within each model type is small, showing that the models are robust to the noise in previous utterances. Note that the four conversational models use a vector to encode the whole sequence of utterances rather than each utterance separately, so they are less affected by noise in individual utterances.

8 Conclusions

In this paper, we explore the task of detecting online conversation derailment. We take a broader view of derailment and focus on conversation derailment based on topic changes. We create a new multimodal Conversation Derailment (MCD) dataset, which introduces multimodality into conversation derailment detection. Furthermore, we propose a novel hierarchical architecture – MMConv – that uses visual and conversational contexts. Our experiments show that the proposed model outperforms strong baselines in conversation derailment detection. Finally, we perform a robustness analysis by evaluating models with noisy texts. We find that models using visual and conversational information have increased robustness, which can help build more accurate practical moderation applications. Future work in this direction could involve extending the size of the MCD dataset to include more topics in texts/images, building pre-trained multimodal conversational models for the multimodal derailment detection task, and applying distillation to MMConv model to reduce model size.

9 Limitations

The main limitation of this work is the requirement of large GPU resources. Note that the proposed dataset is sourced from Reddit where user-generated texts can be very long, the model requires larger GPU memory to process the long inputs. In the MCD dataset, the longest target text contains 1,678 words and the longest conversation contains 56 previous utterances. To fit the long inputs into the GPU, we have to truncate the previous utterances and the target text, which might cause information to be lost. In addition, in the proposed MMConv model, pre-trained vision models such as ViT, are not used because of the large cost of GPU resources compared to using pre-extracted object features. We also use gradient accumulation to perform large-batch training. These trade-offs to fit the model in smaller GPU usage limit the model performance. In addition, another potential limitation is that the annotation agreement between crowd-workers is not perfect. Finally, a potential limitation of the data could be the size/diversity of the dataset, where the texts might represent a specific domain. A possible extension would be to increase the number of subreddits to improve the diversity of the covered topics.

10 Ethical Considerations

In this paper, we study the task of detecting online conversation derailment. We create a new multimodal Conversation Derailment dataset. Our data is collected from Reddit using the PushShift API and Reddit API. The data is annotated by crowd-workers from Amazon Mechanical Turk. Before the annotation, we collect informed consent that the annotation will be used for research in conversations derailment. Given that the data contains a small proportion of toxic language, we require the annotators to pass the adult qualification and state clearly in the annotation interface that the workers can exit at any point, i.e. there is no minimal number of samples to be completed for payment. No personal information of the workers is collected, and the dataset does not include annotators' IDs.

We propose a model architecture and perform experiments on conversation derailment detection. The technology can be potentially used in online social platforms as an auto-moderator tool. Therefore, human moderators could benefit from the technology by spending less time on moderation. In failure cases, false negative might cause failed moderation, thus allowing unwanted language to remain in the platform. We are aware that the current size of data and model might make them potentially biased to certain types of images and topics. Expanding the size of the dataset might help mitigate such biases.

References

Hind Almerekhi, Supervised by Bernard J. Jansen, and co-supervised by Haewoon Kwak. 2020. *Investigating Toxicity Across Multiple Reddit Communities, Users, and Moderators*, page 294–298. Association for Computing Machinery, New York, NY, USA.

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*.

Jason Baumgartner, Savvas Zannettou, Brian Keegan, Megan Squire, and Jeremy Blackburn. 2020. The pushshift reddit dataset.

Christos Baziotis, Nikos Pelekis, and Christos Doulkeridis. 2017. Datastories at semeval-2017 task 4: Deep lstm with attention for message-level and topic-based sentiment analysis. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 747–754, Vancouver, Canada. Association for Computational Linguistics.

Giorgos Bouritsas, Petros Koutras, Athanasia Zlatintsi, and Petros Maragos. 2018. Multimodal visual concept learning with weakly supervised techniques. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Noé Cécillon, Vincent Labatut, Richard Dufour, and Georges Linarès. 2020. WAC: A corpus of Wikipedia conversations for online abuse detection. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 1382–1390, Marseille, France. European Language Resources Association.

Jonathan P. Chang and Cristian Danescu-Niculescu-Mizil. 2019. Trouble on the horizon: Forecasting the derailment of online conversations as they develop. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4743–4754, Hong Kong, China. Association for Computational Linguistics.

Ming-Wei Chang, Kristina Toutanova, Kenton Lee, and Jacob Devlin. 2019. Language model pre-training for hierarchical document representations.

Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, Doha, Qatar. Association for Computational Linguistics.

Elizabeth F. Churchill and Sara Bly. 2000. Culture vultures: Considering culture and communication in virtual environments. *SIGGROUP Bull.*, 21(1):6–11.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Kaustubh D. Dhole, Varun Gangal, Sebastian Gehrmann, Aadesh Gupta, Zhenhao Li, Saad Mahamood, Abinaya Mahendiran, Simon Mille, Ashish Srivastava, Samson Tan, Tongshuang Wu, Jascha Sohl-Dickstein, Jinho D. Choi, Eduard Hovy, Ondrej Dusek, Sebastian Ruder, Sajant Anand, Nagender Aneja, Rabin Banjade, Lisa Barthe, Hanna Behnke, Ian Berlot-Attwell, Connor Boyle, Caroline Brun, Marco Antonio Sobrevilla Cabezudo, Samuel Cahyawijaya, Emile Chapuis, Wanxiang Che, Mukund Choudhary, Christian Clauss, Pierre Colombo, Filip Cornell, Gautier Dagan, Mayukh Das, Tanay Dixit, Thomas Dopierre, Paul-Alexis Dray, Suchitra Dubey, Tatiana Ekeinhor, Marco Di Giovanni, Rishabh Gupta, Rishabh Gupta, Louanes Hamla, Sang Han, Fabrice Harel-Canada, Antoine Honore, Ishan Jindal, Przemyslaw K. Joniak, Denis Kleyko, Venelin Kovatchev, Kalpesh Krishna, Ashutosh Kumar, Stefan Langer, Seungjae Ryan Lee, Corey James Levinson, Hualou Liang, Kaizhao Liang, Zhexiong Liu, Andrey Lukyanenko, Vukosi Marivate, Gerard de Melo, Simon Meoni, Maxime Meyer, Afnan Mir, Nafise Sadat Moosavi, Niklas Muennighoff, Timothy Sum Hon Mun, Kenton Murray, Marcin Namysl, Maria Obedkova, Priti Oli, Nivranshu Pasricha, Jan Pfister, Richard Plant, Vinay Prabhu, Vasile Pais, Libo Qin, Shahab Raji, Pawan Kumar Rajpoot, Vikas Raunak, Roy Rinberg, Nicolas Roberts, Juan Diego Rodriguez, Claude Roux, Vasconcellos P. H. S., Ananya B. Sai, Robin M. Schmidt, Thomas Scialom, Tshephisho Sefara, Saqib N. Shamsi, Xudong Shen, Haoyue Shi, Yiwen Shi, Anna Shvets, Nick Siegel, Damien Sileo, Jamie Simon, Chandan Singh, Roman Sitelew, Priyank Soni, Taylor Sorensen, William Soto, Aman Srivastava, KV Aditya Srivatsa, Tony Sun, Mukund Varma T, A Tabassum, Fiona Anting Tan, Ryan Teehan, Mo Tiwari, Marie Tolkiehn, Athena Wang, Zijian Wang, Gloria Wang, Zijie J. Wang, Fuxuan Wei, Bryan Wilie, Genta Indra Winata, Xinyi Wu, Witold Wydmański, Tianbao Xie, Usama Yaseen, M. Yee, Jing Zhang, and Yue Zhang. 2021. Nl-augmenter: A framework for task-sensitive natural language augmentation.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*.

Desmond Elliott. 2018. Adversarial evaluation of multimodal machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2974–2978, Brussels, Belgium. Association for Computational Linguistics.

Xiaodong Gu, Kang Min Yoo, and Jung-Woo Ha. 2021. Dialogbert: Discourse-aware response generation via learning to recover and rank utterances. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(14):12911–12919.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

Dan Hendrycks and Kevin Gimpel. 2016. Gaussian error linear units (gelus).

Julia Ive, Atijit Anuchitanukul, and Lucia Specia. 2021. Revisiting contextual toxicity detection in conversations.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2020. Is bert really robust? a strong baseline for natural language attack on text classification and entailment. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8018–8025.

Yova Kementchedjhieva and Anders Søgaard. 2021. Dynamic forecasting of conversation derailment. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7915–7919, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Douwe Kiela, Suvrat Bhooshan, Hamed Firooz, and Davide Testuggine. 2019. Supervised multimodal bitransformers for classifying images and text. In *Visually Grounded Interaction and Language (ViGIL), NeurIPS 2019 Workshop, Vancouver, Canada, December 13, 2019*.

Wonjae Kim, Bokyung Son, and Ildoo Kim. 2021. Vilt: Vision-and-language transformer without convolution or region supervision. In *ICML*, pages 5583–5594.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Tuan Manh Lai, Quan Hung Tran, Trung Bui, and Daisuke Kihara. 2020. A simple but effective bert model for dialog state tracking on resource-limited systems. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8034–8038.

Charlotte Lambert, Ananya Rajagopal, and Eshwar Chandrasekharan. 2022. Conversational resilience: Quantifying and predicting conversational outcomes following adverse events. *Proceedings of the International AAAI Conference on Web and Social Media*, 16(1):548–559.

Tianda Li, Jia-Chen Gu, Xiaodan Zhu, Quan Liu, Zhen-Hua Ling, Zhiming Su, and Si Wei. 2020. Dialbert: A hierarchical pre-trained model for conversation disentanglement.

Zhenhao Li, Marek Rei, and Lucia Specia. 2021. Visual cues and error correction for translation robustness. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3153–3168, Punta Cana, Dominican Republic. Association for Computational Linguistics.

John Morris, Eli Lifland, Jin Yong Yoo, Jake Grigsby, Di Jin, and Yanjun Qi. 2020. Textattack: A framework for adversarial attacks, data augmentation, and adversarial training in nlp. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 119–126.

Raghavendra Pappagari, Piotr Zelasko, Jesús Villalba, Yishay Carmiel, and Najim Dehak. 2019. Hierarchical transformers for long document classification. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 838–844.

Chan Young Park, Julia Mendelsohn, Karthik Radhakrishnan, Kinjal Jain, Tushar Kanakagiri, David Jurgens, and Yulia Tsvetkov. 2021. Detecting community sensitive norm violations in online conversations. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3386–3397, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

Bishal Santra, Potnuru Anusha, and Pawan Goyal. 2021. Hierarchical transformer for task oriented dialog systems. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5649–5658, Online. Association for Computational Linguistics.

Tamara Shepherd, Alison Harvey, Tim Jordan, Sam Srauy, and Kate Miltner. 2015. Histories of hating. *Social Media + Society*, 1(2):2056305115603997.

Alessandro Sordoni, Yoshua Bengio, Hossein Vahabi, Christina Lioma, Jakob Grue Simonsen, and Jian-Yun Nie. 2015. 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*, CIKM '15, page 553–562, New York, NY, USA. Association for Computing Machinery.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Bertie Vidgen, Dong Nguyen, Helen Margetts, Patricia Rossini, and Rebekah Tromble. 2021. Introducing CAD: the contextual abuse dataset. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2289–2303, Online. Association for Computational Linguistics.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489, San Diego, California. Association for Computational Linguistics.

Justine Zhang, Jonathan Chang, Cristian Danescu-Niculescu-Mizil, Lucas Dixon, Yiqing Hua, Dario Taraborelli, and Nithum Thain. 2018a. 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)*, pages 1350–1361, Melbourne, Australia. Association for Computational Linguistics.

Justine Zhang, Cristian Danescu-Niculescu-Mizil, Christina Sauper, and Sean J. Taylor. 2018b. Characterizing online public discussions through patterns of participant interactions. *Proc. ACM Hum.-Comput. Interact.*, 2(CSCW).

Xingxing Zhang, Furu Wei, and Ming Zhou. 2019. HIBERT: Document level pre-training of hierarchical bidirectional transformers for document summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5059–5069, Florence, Italy. Association for Computational Linguistics.

Ming Zhong, Yang Liu, Yichong Xu, Chenguang Zhu, and Michael Zeng. 2021. Dialoglm: Pre-trained model for long dialogue understanding and summarization.

A Annotator Qualification Process

Workers must pass the all of the following qualifications before they are allowed to annotate the data:

- The worker must be an **adult** since the data contains potential offensiveness.
- The worker's **number of approved annotations** must be above 2,000 on AMT platform.
- The worker's **approval rate** must be above 95% on AMT platform.
- The worker must pass a **qualification test** of 10 random conversations from the gold data and achieve an accuracy above 80%.

The worker is granted access to be the qualified worker. Note that we also perform quality control during the annotation and block annotators with poor annotations. Once a worker is blocked, the work is no longer able to annotate more data.

B Annotation Interface

An example of the annotation interface is shown in Figure 6.

C MMConv with different pre-trained encoders

In Table 5, the performance of MMConv model with different pre-trained encoders is presented. We notice that BERT performs better than the other two pre-trained language models, i.e. RoBERTa and XLNet. Such improvement might be introduced by the next-sentence-prediction objective used in BERT pre-training, which might help conversational coherence modelling. In terms of visual encoders, BUTD features outperform ViT by a small margin.

Text	Visual	Acc	Pre	Rec	F1
BERT	BUTD	.756	.704	.724	.714
BERT	ViT	.753	.703	.711	.707
RoBERTa	BUTD	.729	.671	.683	.677
RoBERTa	ViT	.715	.658	.662	.660
XLNet	BUTD	.729	.653	.754	.700
XLNet	ViT	.725	.672	.673	.673

Table 5: Performance of MMConv model with different pre-trained text and visual encoders.

D Forecast Horizon Distribution for MMConv

Figure 7 shows the distribution of forecast horizon for MMConv model in predicting future comments. The model does not predict future derailment on 25% of true positive examples, but could give early signal on future derailments within 4 utterances for over 40% of the cases. Finally, the model is also capable of handling long conversations with over 13 utterances and predict future derailment at an early stage.

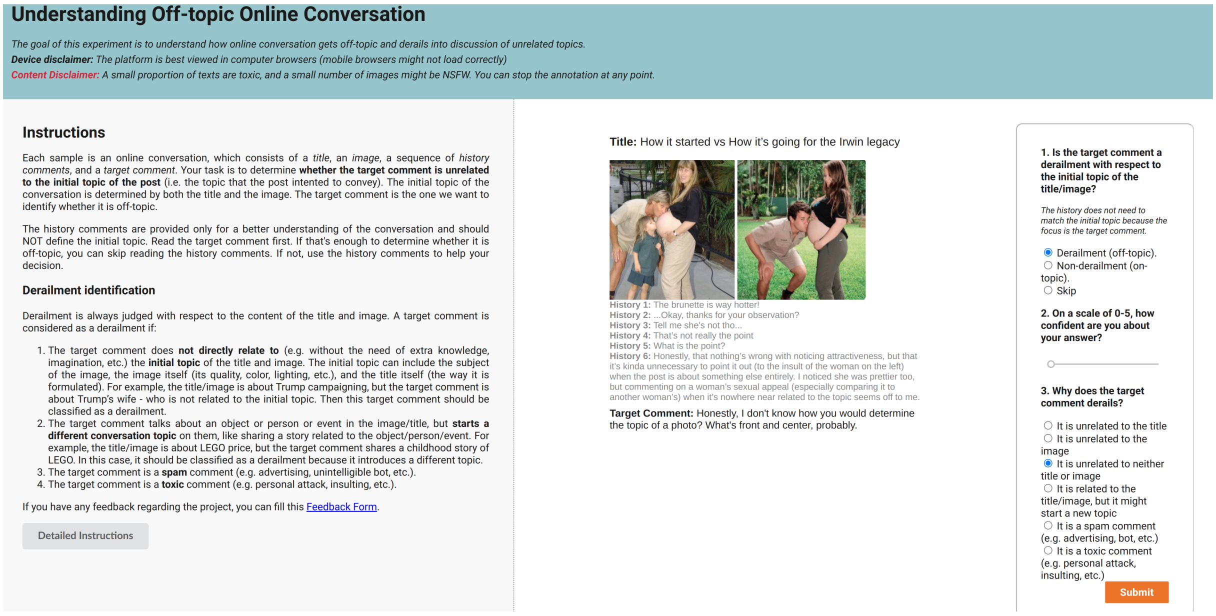
E Noise Injection Strategies

We used seven common noise injection strategies from NL-Augmenter:

- abbreviation transformation: words are randomly replaced with their abbreviations if the abbreviations exist.
- butter fingers perturbation: characters are randomly replaced with adjacent characters on a standard QWERTY keyboard.
- close homophones swap: words are randomly swapped to another word that look or sound similarly with the original word.
- multilingual dictionary based code switch: words are randomly replaced with its translation in another language.
- visual attack letters: characters are randomly substituted with another character (might from a different encoding) that looks similar.
- whitespace perturbation: whitespaces are randomly inserted or deleted in the text.

F Robustness to attack

The other aspect of robustness is on adversarial attacks. Although the model shows robustness on



Understanding Off-topic Online Conversation

The goal of this experiment is to understand how online conversation gets off-topic and derails into discussion of unrelated topics.

Device disclaimer: *The platform is best viewed in computer browsers (mobile browsers might not load correctly)*

Content Disclaimer: *A small proportion of texts are toxic, and a small number of images might be NSFW. You can stop the annotation at any point.*

Instructions

Each sample is an online conversation, which consists of a *title*, an *image*, a sequence of *history comments*, and a *target comment*. Your task is to determine **whether the target comment is unrelated to the initial topic of the post** (i.e. the topic that the post intented to convey). The initial topic of the conversation is determined by both the title and the image. The target comment is the one we want to identify whether it is off-topic.

The history comments are provided only for a better understanding of the conversation and should NOT define the initial topic. Read the target comment first. If that's enough to determine whether it is off-topic, you can skip reading the history comments. If not, use the history comments to help your decision.

Derailment identification

Derailment is always judged with respect to the content of the title and image. A target comment is considered as a derailment if:

1. The target comment does **not directly relate to** (e.g. without the need of extra knowledge, imagination, etc.) the **initial topic** of the title and image. The initial topic can include the subject of the image, the image itself (its quality, color, lighting, etc.), and the title itself (the way it is formulated). For example, the title/image is about Trump campaigning, but the target comment is about Trump's wife - who is not related to the initial topic. Then this target comment should be classified as a derailment.
2. The target comment talks about an object or person or event in the image/title, but **starts a different conversation topic** on them, like sharing a story related to the object/person/event. For example, the title/image is about LEGO price, but the target comment shares a childhood story of LEGO. In this case, it should be classified as a derailment because it introduces a different topic.
3. The target comment is a **spam** comment (e.g. advertising, unintelligible bot, etc.).
4. The target comment is a **toxic** comment (e.g. personal attack, insulting, etc.).

If you have any feedback regarding the project, you can fill this Feedback Form.

Title: How it started vs How it's going for the Irwin legacy

History 1: The brunette is way hotter!
History 2: ...Okay, thanks for your observation?
History 3: Tell me she's not tho...
History 4: That's not really the point
History 5: What is the point?
History 6: Honestly, that nothing's wrong with noticing attractiveness, but that it's kinda unnecessary to point it out (to the insult of the woman on the left) when the post is about something else entirely. I noticed she was prettier too, but commenting on a woman's sexual appeal (especially comparing it to another woman's) when it's nowhere near related to the topic seems off to me.

Target Comment: Honestly, I don't know how you would determine the topic of a photo? What's front and center, probably.

1. Is the target comment a derailment with respect to the initial topic of the title/image?

The history does not need to match the initial topic because the focus is the target comment.

● Derailment (off-topic).
○ Non-derailment (on-topic).
○ Skip

2. On a scale of 0-5, how confident are you about your answer?

3. Why does the target comment derails?

○ It is unrelated to the title
○ It is unrelated to the image
● It is unrelated to neither title or image
○ It is related to the title/image, but it might start a new topic
○ It is a spam comment (e.g. advertising, bot, etc.)
○ It is a toxic comment (e.g. personal attack, insulting, etc.)

Figure 6: Annotation interface on Amazon Mechanical Turk.

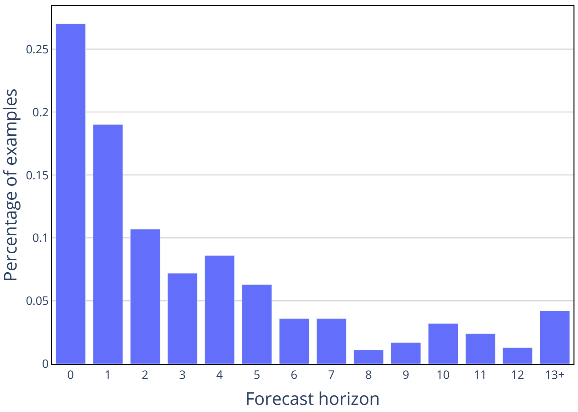


Figure 7: Forecast horizon distribution for MMConv model.

noisy texts, users might escape the auto-moderation by iteratively paraphrasing the target text until the model misclassifies, and such a process can be seen as an adversarial attack. To simulate this, we use TextAttack (Morris et al., 2020) to perform adversarial attacks to the models. We randomly sample 200 examples from the test set and attack the model with *textfooler* attack (Jin et al., 2020). We calculate the successful attack rate and the average number of required queries to measure the model's robustness. If a model is robust, the attack method would have difficulties finding an adversarial example, thus resulting in lower successful attack rate and higher query number. The model performance is presented in Table 6.

Models	Success rate (↓)	Avg query (↑)
BERT-simple	92.8	137
CRAFT	84.6	165
ContextRNN	**75.2**	**196**
ToBERT	79.6	186
MMConv	80.2	188

Table 6: Results of successful attack rate and average query number with textfooler attack (Jin et al., 2020).

The BERT-simple baseline exhibits the least robustness to attack, with a successful attack rate of 92.8% and average query number of 137. Other conversational models, though not optimized with adversarial training, all show obvious improvements over the text-only baseline, indicating that the additional conversational context could increase model robustness. Among all models, ContextRNN shows the best robustness, decreasing the successful attack rate significantly to 75.2%, and increasing the average query number to 196.