Multi-task transfer learning for finding actionable information from crisis-related messages on social media

Congcong Wang
School of Computer Science
University College Dublin
Dublin, Ireland
congcong.wang@ucdconnect.ie

David Lillis
School of Computer Science
University College Dublin
Dublin, Ireland
david.lillis@ucd.ie

Abstract

The Incident streams (IS) track is a research challenge aimed at finding important information from social media during crises for emergency response purposes. More specifically, given a stream of crisis-related tweets, the IS challenge asks a participating system to 1) classify what the types of users' concerns or needs are expressed in each tweet, known as the information type (IT) classification task and 2) estimate how critical each tweet is with regard to emergency response, known as the priority level prediction task. In this paper, we describe our multi-task transfer learning approach for this challenge. Our approach leverages state-of-the-art transformer models including both encoder-based models such as BERT and a sequence-to-sequence based T5 for joint transfer learning on the two tasks. Based on this approach, we submitted several runs to the track. The returned evaluation results show that our runs substantially outperform other participating runs in both IT classification and priority level prediction.

1 Introduction

Social media platforms such as Twitter have made it possible for users to report on an ongoing event in their vicinity in a timely manner (Fraustino et al., 2012). This has motivated researchers to explore the potential of social media platforms for finding actionable information from this user-generated content during a crisis event (Caragea et al., 2011; Imran et al., 2015; McCreadie et al., 2019). Finding this type of information is especially important for emergency response agencies to enable them to take immediate actions to help those who are posting for help, which is known as situational awareness (Vieweg, 2012; Vieweg et al., 2010). This naturally raises the question: how can the process of finding the actionable information effectively be automated, given the fact that the messages posted during a crisis on social media are usually noisy and numerous?

The Incident streams (IS) track (McCreadie et al., 2019, 2020) is proposed by the **T**ext **RE**trieval **C**onference (TREC) as a research challenge for this purpose. Since it was introduced in 2018, the IS track has conducted two major tasks regarding crisis short message processing. Given a stream of tweets from crisis events, the foremost task is that it asks a participating system to classify the information types (ITs) for each tweet. The ITs are simply a pre-defined set of classes in relation to something that a user is likely to post during a crisis. The ITs can be something important such as *requesting research and rescue, call for moving people, reporting goods available*, etc., as well as something less important such as *reporting weather or location, expressing sentiment*, etc.[1] In addition to the ITs classification task, the IS track also asks the participating systems to estimate the priority level for each tweet, indicating how important the tweet is in taking immediate emergency response actions. The IS track predefines four priority levels: *critical*, *high*, *medium* and *low*, which are ordered from the highest to lowest priority.

The IS track was run once in 2018 and twice in each subsequent year, so it has accumulated five editions as of 2020. For each edition, an annotated collection of tweets from previous editions is used as the training data for the community, and unseen tweets (non-annotated) are released as the test tweets for official evaluation. The two most recent editions, conducted in 2020, are named 2020A and 2020B respectively. Slightly different from previous editions, the two editions introduce a reduced set of ITs as well as a set of test tweets related to

[1] There are 6 important ITs known as "actionable" ITs predefined by the IS track and 19 are considered to be "non-actionable". For details, see (McCreadie et al., 2019).

the COVID-19 pandemic, resulting in three tasks described as follows.

- **Task 1**: This task remains the same as the editions before 2020, it uses all 25 ITs for classification and four priority levels for estimation.
- **Task 2**: Different from Task 1, this task only asks the participating systems to classify one or more of 12 IT classes. The 12 ITs include 11 that are closely related to emergency response and the remaining as "Other-Any" [2].
- **Task 3**: Unlike Task 1 and 2 that relate to general crises such as earthquakes, explosions or hurricanes, this task focuses on the COVID-19 domain. It provides a stream of COVID-related tweets from different locations for IT classification using only a subset of 9 ITs suitable for COVID-19 and priority estimation using the same four priority levels as used in Task 1 and 2.

In this paper, we describe our system's approach in the three tasks of the IS track from our participation in both 2020A and 2020B. For different tasks, we submitted different runs but all were based on the multi-task transfer learning approach that we utilised in our system. Given the recent success of transformers (Wolf et al., 2020) in transfer learning for various language tasks such as sentence classification, question answering, etc., we leverage them in the IS challenge. We explored transformer encoder based models such as BERT (Devlin et al., 2019) and a sequence-to-sequence model - T5 (Raffel et al., 2020) for their potential in this challenge. By doing so, we fine-tune them in a multi-task learning fashion (i.e, joint fine-tuning of the IT classification and priority estimation). With this approach, we submitted five runs to the IS track. The evaluation results show that our runs substantially outperform other participating runs in both IT classification and priority level prediction.

2 Related Work

To improve emergency response, the community has seen many works on exploring computational techniques for knowledge acquisition from crisis messages on social media. Caragea et al. (2011) applied traditional machine learning algorithms including LDA and SVM to find important information such as *people trapped* or *food shortage* from the 2010 Haiti Earthquake. As neural network (NN) approaches have gained popularity in recent years, many deep learning approaches have been applied to this domain. For example, Nguyen et al. (2017) applied a convolution neural network (CNN) for classifying informative tweets from general disasters such as the *2015 Nepal Earthquake*, *Typhoon Hagupit*, etc., whereas Alam et al. (2018) leveraged a CNN with adversarial training for identifying whether a tweet is relevant to a certain crisis event.

In recent years, since the attention-based transformer model was introduced (Vaswani et al.), several variations have been proposed such as BERT (Devlin et al., 2019), ELECTRA (Clark et al., 2020) and T5 (Raffel et al., 2020), collectively known as the transformers (Wolf et al., 2020), achieving state-of-the-art performance in many language tasks with transfer learning. It is common that the transformers are first pre-trained on a large general text corpus and then are fine-tuned on specific downstream language tasks such as text classification. Given the strong transfer capability of transformers, they have been widely studied for crisis messages processing also. Liu et al. (2020) fine-tuned BERT for crisis identification and detection tasks and Wang and Lillis (2020b) applied T5 for extracting useful information such as *who tested positive/negative or cannot get test* from COVID-related tweets by treating it as a question-answering task. Our approach in the IS track is similar to this line of work, which applies the transformers with transfer learning for finding actionable information in the tasks as proposed by the IS track. However, our approach is different in the way it fine-tunes the transformers by multi-task learning, aiming to make use of shared model weights between different tasks.

Since the IS track has been run for several years, the participating systems have proposed various techniques specifically for this track. Such approaches can broadly be summarised in three categories. First, traditional machine learning algorithms have been used with careful pre-processing steps and handcrafted input features. For example, Wang et al. applied models including Naïve Bayes, SVM, Random Forest, and the ensemble

[2]For full details, refer to http://dcs.gla.ac.uk/~richardm/TREC_IS/2020/participate.html

of these models. To train these models, they used hand-crafted features such as the length, sentiment polarity of a tweet, number of followers of the user, combining with context-free GloVe and FastText embeddings as well as context-aware BERT embeddings as the input features. The second category uses deep learning approaches that pre-date the widespread adoption of transformers. For instance, Miyazaki et al. (2019) proposed the method using label embedding with a BiLSTM model in this track while Wang and Lillis (2020a) applied a BiLSTM network along with pre-trained ELMo embeddings and trainable embeddings as the input features for crisis tweet categorisation. The last category encompasses transformer-based fine-tuning approaches. One example is that Zahera et al. (2019) fine-tuned BERT for the multi-label ITs classification task using the training tweets after preprocessing.

3 Method

Our approach is based on multi-task transfer learning through fine-tuning both transformer encoder-based models such as BERT and sequence-to-sequence transformers such as T5. The following details the process of the two types of models used, which we name the **encoders scenario** and **sequence-to-sequence scenario** respectively. Each type of model was used for both the IT classification task and priority prediction task.

Encoders scenario: This scenario simply adds two linear projection layers on top of transformer encoders such as BERT. Our architecture is agnostic as to the specific transfer encoder used. One projection layer transforms the encoder's pooled output (namely, the `[CLS]` output vector of BERT) to a vector representing the IT classes. The IT representation is then passed to the *sigmoid* function that calculates the probability distribution for every IT class. The other projection layer is used to transform the encoder's output to a vector representing the four priority levels. Similarly, it is then passed to the *sigmoid* function, which calculates a score indicating the priority levels as follows.

$$
\begin{aligned}
(0.75, 1] &\longrightarrow \textbf{Critical} \\
(0.5, 0.75] &\longrightarrow \textbf{High} \\
(0.25, 0.5] &\longrightarrow \textbf{Medium} \\
[0.0, 0.25] &\longrightarrow \textbf{Low}
\end{aligned}
\tag{1}
$$

In order to achieve the joint learning of both tasks, the encoder model is fine-tuned with the loss function linearly combining the binary cross entropy loss between the IT probability distribution and ground truths (a multi-label classification problem) as well as the mean squared error between the importance scores and priority ground truths (a regression problem).

Sequence-to-sequence scenario (seq2seq): This scenario is mostly motivated by the work that applies T5 for COVID-related event extraction by treating it as a multi-choice question answering task (Wang and Lillis, 2020a). We adapt it to the IS track for multi-task transfer learning using seq2seq transformers such as T5. Basically, the seq2seq model takes a sequence of text as the input, known as the source sequence, and outputs the target sequence conditional on the source sequence. Under this mechanism, the template used to construct the source and target sequences in both tasks of the IS track is presented as follows.

Source: context: T question: IQ/PQ choices: IC/PC
Target: I/P

- **T** refers to the raw tweet text without any reprocessing except for being lower-cased.
- **IQ/PQ** refers to the IT classification and priority estimation task-specific ad-hoc question texts, which are "**what type of information does the tweet convey relating to a crisis?**" and "**what level of urgency is likely expressed in this tweet relating to a crisis?**" respectively.
- **IC/PC** implies the flatted texts concatenating all IT and priority levels respectively. For example, IC is something like "**call for donations, call to move people, ...**" which varies in different IT classification tasks. The PC is simply "**critical, high, medium, low**".
- **I/P** indicates the generated predictions for ITs and priority level, which are direct textual predictions from **IC/PC** respectively.

Using this template, each tweet in the training set is converted to an IT-specific source-target pair and a priority-specific source-target pair. In order to achieve the joint learning of both tasks, the sequence-to-sequence model is fine-tuned on batches of training sequences that contain both the IT pairs and priority pairs.

runtag	scenario	task target	submission type	training data
run1	Encoders	Task 1 & 2	one-hot	prior to 2020B excluding COVID
run2	Encoders	Task 1 & 2	probability	prior to 2020B excluding COVID
run3	seq2seq	Task 1 & 2	one-hot	prior to 2020B excluding COVID
run4	seq2seq	Task 3	one-hot	prior to 2020B including COVID
run5	seq2seq	Task 1 & 2	one-hot	prior to 2020B including COVID

Table 1: The summary of our submitted runs for TREC-IS 2020-B. Run1, 2, 3, 5 submitted to task 1 are also submitted to task 2 for evaluation.

4 Experiments

This section describes the details of our system's runs submitted to the latest 2020B edition of the IS track. Since our system was developed based on our previous experience in this track, the method we described in Section 3 also covers our approach to the 2020A edition (actually the **encoders scenario**). Our baseline run (run1) for 2020B, which is an ensemble run under the **encoders scenario** from 2020A that we consider as a strong baseline. In 2020B, we submitted a total of five runs to Task 1, 2 and 3 as mentioned in Section 1 and they are summarised in Table 1 and described as follows.

- **run1**: This is a baseline with techniques initially developed in 2019A. In 2020A, we proposed the **encoders scenario**, achieving strong performance as compared to other participating techniques. To further make it a strong baseline, we used a simple ensemble approach combining the predictions made by the fine-tuned individual models[3] under the encoders scenario. The ensemble run simply predicts the final IT predictions for each tweet to be the union of individual IT predictions and the final priority level to the highest of the individual priority predictions. Per the guideline of 2020B, both the IT and priority levels are expected to be numeric instead of being categorical as required prior to 2020B. Hence, we transform the final IT predictions to one-hot encodings and map the priority level prediction to its importance score by: *Critical: 1.0, High: 0.75, Medium: 0.5, Low: 0.25*.

- **run2**: Similar to run1, the difference is that for run2, the final ITs predictions are the highest probability values among the predictions by individual models. The final priority predictions are simply the highest of the individual models' outputs without applying the conversion as defined in Equation 1.

- **run3**: For this run, the **seq2seq scenario** is conducted for multi-task transfer learning. We follow the T5 base architecture initialised with `t5-base` weights and fine-tune it on the training tweets prior to 2020B (excluding the COVID-related tweets from the 2020A edition). Since the seq2seq model outputs the generated texts as the predictions for both priority and ITs, we convert the IT predictions to one-hot encodings and priority level to the importance score before they are submitted.

- **run4**: With a similar setup to run3, run4 is submitted for Task 3 and thus it includes the training tweets prior to 2020B including the COVID-related tweets from 2020A.

- **run5**: With a similar setup to run3, run5 is submitted for Task 1 & 2 and it uses all previous training tweets including the COVID tweets for fine-tuning the T5 model.

4.1 Training Details

As described, our runs mainly focus on fine-tuning several transformer encoder models and a **t5-base** sequence-to-sequence model in a multi-task learning way. For the fine-tuning of **t5-base**, we follow the same hyper-parameter configuration as used in Wang and Lillis (2020b). For fine-tuning each of the transformer encoder models, we use the same set of the hyper-parameters that are configured with reference to a similar work in this domain (Liu et al., 2020). For training, we sample around 10% of the training data as

[3]The individual models that were used in this run included fine-tuned `bert-base-uncased`, `electra-base-discriminator`, `albert-base-v2` and `distilbert-base-uncased`, which are all available in the transformers library (Wolf et al., 2020).

Run	nDCG@100	Info-Type F1 [Actionable]	Info-Type F1 [All]	Info-Type Accuracy	Priority F1 [Actionable]	Priority F1 [All]
BJUT-run	0.4346	0.0266	0.0581	0.8321	0.1744	0.0905
njit.s1.aug	0.4480	0.2634	0.3103	**0.8655**	0.2029	0.1518
njit.s2.cmmd.t1	0.4475	0.1879	0.2223	0.8475	0.2029	0.1518
njit.s3.img.t1	0.4222	0.1879	0.2223	0.8475	0.1959	0.1417
njit.s4.cml.t1	0.4164	0.1712	0.1465	0.8445	0.1054	0.1064
ufmg-sars-test	0.3634	0.0001	0.0493	0.8337	0.1285	0.1378
ucd-run1 (ours)	0.5033	**0.3215**	**0.3810**	0.8520	0.2582	0.2009
ucd-run2 (ours)	0.5022	0.3078	0.3692	0.8316	0.2582	0.2016
ucd-run3 (ours)	0.5038	0.3001	0.3448	0.8653	**0.2803**	0.3046
ucd-run5 (ours)	**0.5252**	0.3036	0.3444	0.8601	0.2801	**0.3126**

Table 2: Evaluation results of participating runs at TREC-IS 2020-B Task 1. Highest in columns are bold.

Run	nDCG@100	Info-Type F1 [All]	Info-Type Accuracy	Priority F1 [All]
		Task-1 Systems		
BJUT-run	0.4350	0.0472	0.7977	0.1337
njit.s1.aug	0.4487	0.3480	0.8846	0.1838
njit.s2.cmmd.t1	0.4467	0.2494	0.8612	0.1838
njit.s3.img.t1	0.4215	0.2494	0.8612	0.1708
njit.s4.cml.t1	0.4176	0.1278	0.8360	0.1162
ufmg-sars-test	0.3630	0.0127	0.8419	0.1480
ucd-run1 (ours)	0.5020	**0.4036**	0.8913	0.2320
ucd-run2 (ours)	0.5027	0.3961	0.8364	0.2322
ucd-run3 (ours)	0.5032	0.3689	**0.8932**	0.2867
ucd-run5 (ours)	**0.5240**	0.3674	0.8845	**0.3003**
		Task-2 Systems		
njit.s1.aug.t2	0.4478	0.2548	0.8656	0.1838
njit.s2.cmmd.t2	0.4478	0.2548	0.8656	0.1838
njit.s3.img.t2	0.4213	0.2548	0.8656	0.1708
njit.s4.cml.t2	0.4189	0.1713	0.8327	0.1162
ufmg-sars-test-t2	0.3637	0.0127	0.8419	0.1480

Table 3: Evaluation results of participating runs at TREC-IS 2020-B Task 2. The Task-1 systems refer to the runs from Task 1 re-evaluated under Task 2 while Task-2 systems are the submitted runs specific to Task 2.

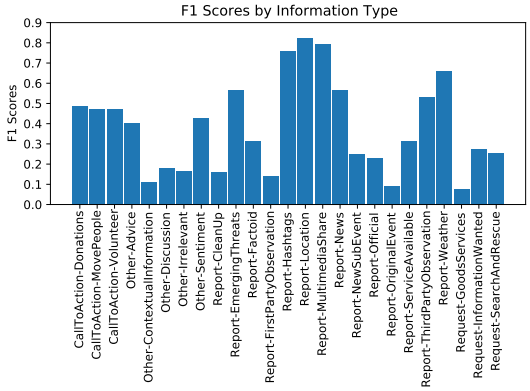
the validation set first. Then, we fine-tune each model with a batch size of 32, learning rate of 5e-5, linear warm-up ratio of 0.1 with Adam optimizer (Kingma and Ba, 2015). For the input length, we set the maximum input length to be 256 since we found few examples has length beyond this number. All training examples in our experiments are not pre-processed but used in raw texts.

4.2 Results

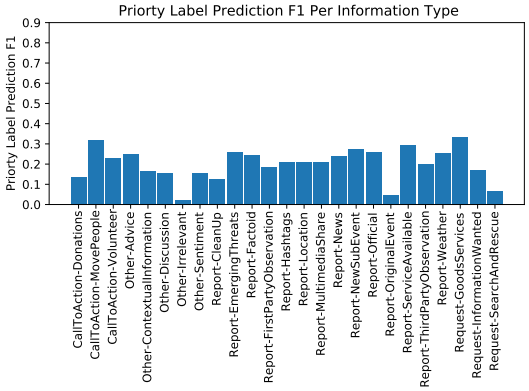
Having submitted the five runs as described in Table 1 to the track, they were officially evaluated and the results are reported in Tables 2, 3 and 4. The tables show the performance of participating runs in Task 1, 2 and 3 respectively. The columns are the official metrics used to evaluate different aspects of a run's performance, which are described briefly as follows.

- **Information type classification**: There are two types of information type (IT) F1. The "Actionable IT" F1 reflects a run's performance in classifying actionable ITs [4]. The "All IT" F1 measures a run's performance across all information types (25 in Task 1, 12 in Task 2 and 9 in Task 3). The IT accuracy is the overall accuracy in IT classification.

[4] They are Request-GoodsService, Request-SearchAndRescue, Report-NewSubEvent, Report-ServiceAvailable, CallToAction-MovePeople, and Report-EmergingThreats.



(a) F1 Scores by Information Type



(b) Priority level prediction F1 per information type

Figure 1: Performance visualisation by information types of **ucd-run1** in Task 1.

Run	nDCG@100	Info-Type F1 [Actionable]	Info-Type F1 [All]	Info-Type Accuracy	Priority F1 [Actionable]	Priority F1 [All]
njit.s1.aug.t3	0.4322	**0.1629**	0.1450	0.8593	0.2551	0.1499
njit.s2.cmmd.t3	0.4329	0.1590	0.1184	0.8586	0.2551	0.1499
njit.s3.img.t3	0.3986	0.1590	0.1184	0.8586	0.2544	0.1562
njit.s4.cml.t3	0.4249	0.0210	0.0650	**0.8626**	0.1375	0.1502
ucd-run4	**0.4497**	0.1425	**0.1817**	0.8541	**0.3443**	**0.2867**

Table 4: Evaluation results of participating runs at TREC-IS 2020-B Task 3.

- **Prioritisation**: Similarly, the Actionable priority F1 measures a run's performance in priority level prediction for only the tweets that are labeled as actionable ITs while the All F1 measures the performance for all test tweets. Moreover, the nDCG@100 is used to measure a run's average performance in ranking top 100 test tweets per event by priority.

As seen from Table 2, in Task 1, our runs substantially outperform other participating runs in both IT classification and prioritisation [5]. In particular, our runs are effective in classifying actionable ITs. For example, our run1 and run3 achieve the top actionable IT F1 score of 0.3215 and the best actionable priority F1 of 0.2803 respectively. This is further evidenced by the runs' performance in Task 2, as in Table 3. All the runs overall perform well in IT classification and prioritisation in Task 2 (the condensed more emergency response related 12 ITs).

In Task 1 and 2, run1 and run2 perform similarly across the metrics since both are based on the encoder scenario and only differ in the final submission type. It is interesting that run5 performs similarly to run3 across the metrics except for being better in nDCG@100: 0.5252 versus 0.5038. The two runs are both based on the seq2seq scenario and only difference is in their training data. This indicates that adding the COVID data (similar domain) to the general crisis data for training can be helpful in the priority-centric ranking performance. To compare between the four runs, it is found that no one run dominates the other runs across all the metrics. This indicates that the multi-task transfer learning approach using either the transformer encoder or the seq2seq as the base model is likely to bring similar performance.

To further examine our runs' performance at every IT level, we report the IT F1s and priority F1s per IT of the run1 in Task 1, as presented in Figure 1. Figure 1a shows that the run performs well in categorising some actionable ITs, such as "CallToAction-MovePeople" and "Report-EmergingThreats" while not the best in actionable ITs such as "Request-GoodsService", as compared to the non-actionable ITs. However, taking a look at the priority F1s per IT in Figure 1b, we found that the run performs relatively better in priority level prediction for actionable ITs than non-actionable ITs,

[5]The exception if accuracy, where only a small difference is observed accuracy across the participating runs: our results are substantially higher than other participating runs in the remaining metrics.

where "CallToAction-MovePeople", "Request-GoodsService" and "Report-ServiceAvailable" are the top 3 the runs achieves in priority F1.

Apart from the four runs to Task 1 and Task 2, we submitted run4 to Task 3 and the results are reported in Table 4. We see that the run is competitive with other participating runs, particularly in prioritisation. Unlike our other four runs in Task 1 and 2, this run achieves 0.1425 in actionable IT F1, next to the best 0.1629. Since Task 3 is COVID-related and newly introduced, we expect our run to be improved in future iterations of this track as more data accumulates.

5 Conclusion

This paper introduces University College Dublin's (UCD) participation in the 2020 TREC-IS track. The IS track was run twice in 2020: namely 2020A and 2020B. Based on our experience from previous editions, we describe our multi-task transfer learning approach using pre-trained encoder-based and sequence-to-sequence transformers. With these approaches, we submitted five runs to the track's 2020-B edition - four for Task 1 and Task 2, and one for Task 3. The results show that our runs to Task 1 and Task 2 substantially outperform other participating runs in both information type classification and priority level prediction. In addition, our runs are effective in finding some actionable information types in Task 1 and Task 2 and the run to Task 3 performs competitively with other participating runs. Regarding future work, we expect to explore the incorporation of knowledge graphs to enhance the model's identification of the crisis-related tweets.

References

Firoj Alam, Shafiq Joty, and Muhammad Imran. 2018. Domain adaptation with adversarial training and graph embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1077–1087.

Cornelia Caragea, Nathan J McNeese, Anuj R Jaiswal, Greg Traylor, Hyun-Woo Kim, Prasenjit Mitra, Dinghao Wu, Andrea H Tapia, C Lee Giles, Bernard J Jansen, et al. 2011. Classifying text messages for the haiti earthquake. In *Proceedings of the 8th International Conference on Information Systems for Crisis Response and Management (ISCRAM 2011)*. Citeseer.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*.

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.

Julia Daisy Fraustino, Brooke Liu, and Yan Jin. 2012. Social media use during disasters: a review of the knowledge base and gaps. *National Consortium for the Study of Terrorism and Responses to Terrorism*.

Muhammad Imran, Carlos Castillo, Fernando Diaz, and Sarah Vieweg. 2015. Processing social media messages in mass emergency: A survey. *ACM Computing Surveys (CSUR)*, 47(4):67.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference for Learning Representations*.

Junhua Liu, Trisha Singhal Lucienne Blessing, Kristin L Wood, and Kwan Hui Lim. 2020. CrisisBERT: Robust Transformer for Crisis Classification and Contextual Crisis Embedding. *arXiv preprint arXiv:2005.06627*.

Richard McCreadie, Cody Buntain, and Ian Soboroff. 2019. TREC incident streams: Finding actionable information on social media. *Proceedings of the International ISCRAM Conference*, 2019-May(May):691–705.

Richard McCreadie, Cody Buntain, and Ian Soboroff. 2020. Incident Streams 2019: Actionable Insights and How to Find Them. In *Proceedings of the 17th International Conference on Information Systems for Crisis Response and Management (ISCRAM 2020)*.

Taro Miyazaki, Kiminobu Makino, Yuka Takei, Hiroki Okamoto, and Jun Goto. 2019. Label embedding using hierarchical structure of labels for twitter classification. 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 6318–6323.

Dat Tien Nguyen, Kamela Ali Al Mannai, Shafiq Joty, Hassan Sajjad, Muhammad Imran, and Prasenjit Mitra. 2017. Robust classification of crisis-related data on social networks using convolutional neural networks. In *Eleventh International AAAI Conference on Web and Social Media*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*.

Sarah Vieweg, Amanda L Hughes, Kate Starbird, and Leysia Palen. 2010. Microblogging during two natural hazards events: what twitter may contribute to situational awareness. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 1079–1088. ACM.

Sarah Elizabeth Vieweg. 2012. *Situational awareness in mass emergency: A behavioral and linguistic analysis of microblogged communications*. Ph.D. thesis, University of Colorado at Boulder.

Congcong Wang and David Lillis. 2020a. Classification for Crisis-Related Tweets Leveraging Word Embeddings and Data Augmentation. In *Proceedings of the Twenty-Eighth Text REtrieval Conference (TREC 2019)*, Gaithersburg, MD.

Congcong Wang and David Lillis. 2020b. UCD-CS at W-NUT 2020 Shared Task-3: A Text to Text Approach for COVID-19 Event Extraction on Social Media. In *Proceedings of the Sixth Workshop on Noisy User-Generated Text (W-NUT 2020)*, pages 514–521. Association for Computational Linguistics.

Junpei Zhou Xinyu Wang, Po-yao Huang, and Alexander Hauptmann. CMU-Informedia at TREC 2019 Incident Streams Track. In *Proceedings of the Twenty-Eighth Text REtrieval Conference (TREC 2019)*, Gaithersburg, MD.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Hamada M Zahera, Ibrahim A Elgendy, Rricha Jalota, and Mohamed Ahmed Sherif. 2019. Fine-tuned BERT Model for Multi-Label Tweets Classification. In *Proceedings of the Twenty-Eighth Text REtrieval Conference (TREC 2019)*, Gaithersburg, MD.