Combining Self-training with Deep Learning for Disaster Tweet Classification

Hongmin Li
Department of Computer Science
Kansas State University
hongminli@ksu.edu

Doina Caragea
Department of Computer Science
Kansas State University
dcaragea@ksu.edu

Cornelia Caragea
Department of Computer Science
University of Illinois at Chicago
cornelia@uic.edu

ABSTRACT

Significant progress has been made towards automated classification of disaster or crisis related tweets using machine learning approaches. Deep learning models, such as Convolutional Neural Networks (CNN), domain adaptation approaches based on self-training, and approaches based on pre-trained language models, such as BERT, have been proposed and used independently for disaster tweet classification. In this paper, we propose to combine self-training with CNN and BERT models, respectively, to improve the performance on the task of identifying crisis related tweets in a target disaster where labeled data is assumed to be unavailable, while unlabeled data is available. We evaluate the resulting self-training models on three crisis tweet collections and find that: 1) the pre-trained language model BERTweet is better than the standard BERT model, when fine-tuned for downstream crisis tweets classification; 2) self-training can help improve the performance of the CNN and BERTweet models for larger unlabeled target datasets, but not for smaller datasets.

Keywords

Domain adaptation, self-training, crisis tweets classification, BERT, Convolutional Neural Network(CNN).

INTRODUCTION

Social media platforms, such as Twitter, are known to have intrinsic value in terms of improving situational awareness during emergencies. Both emergency practitioners and researchers agree that the first hand information posted by people in the affected areas on social media should be integrated into the emergency operations (Homeland Security 2014; Reuter, Hughes, et al. 2018; Castillo 2016; Imran, Castillo, et al. 2015; Palen and Hughes 2018; Grace et al. 2019). However, information accumulates very fast on social media during emergency situations. The information from eyewitnesses or victims of a disaster, can be easily buried deep under the torrent of news reports, unrelated posts or even misinformation. This information overload problem makes it hard for the public to resort to the social media for help, and for the emergency organizations, such as first responders, to improve situational awareness through social media (Plotnick et al. 2015; Reuter, Ludwig, et al. 2015; Imran, Ofli, et al. 2020; Hiltz et al. 2020). Therefore, automated filtering tools are expected to be key in solving this problem (National Research Council 2013; Imran, Ofli, et al. 2020).

In recent years, many researchers have worked on applying machine learning and natural language processing (NLP) techniques to build automated filtering tools to classify disaster related information, categorize sources and information types, and recognize rumors on social media (Imran, Castillo, et al. 2015; Castillo 2016). For classifying disaster or crisis related tweets, researches have employed statistical learning or traditional supervised

machine learning approaches (Verma et al. 2011; Starbird et al. 2010; Imran, Elbassuoni, et al. 2013; C. Caragea, Squicciarini, et al. 2014; Kaufhold et al. 2020; Ghafarian and Yazdi 2020), and deep learning approaches, such as Convolutional Neural Networks (CNN) (D. T. Nguyen, Joty, et al. 2016; D. T. Nguyen, Al-Mannai, et al. 2016; Burel and Alani 2018; Kersten et al. 2019; Ning et al. 2019).

However, to train a good model, many machine learning algorithms, especially the data-hungry deep learning networks, need large amounts of labeled data, which will not be available in a short time for a new emerging event. One solution for this problem is to use historical labeled data from previous events. But as each event is unique in terms of type, location, culture, people involved, etc., and different events may cause different social media responses (Palen and Anderson 2016; Palen and Hughes 2018), supervised classifiers trained on a previous emergency event may not generalize well on a current event in practice (Imran, Castillo, et al. 2015; Imran, Elbassuoni, et al. 2013), especially if the previous and the current disasters are of different types (Wiegmann et al. 2020). Therefore, domain adaptation approaches or approaches that focus on a classifier's generalizability across different disaster types have been proposed.

Among many domain adaptation approaches, one way to increase a classifier's generalizability is to use pre-trained models, such as pre-trained word embedding (H. Li, X. Li, et al. 2018), to transfer knowledge from a large body of general-use unlabeled data. With the big impact of large scale pre-trained language models, approaches that utilize pre-trained language models, such as BERT (Devlin et al. 2018) have also been used to increase a model's generalizability across different disasters (Wiegmann et al. 2020; Ma 2019; Coche et al. 2020; Desai et al. 2020).

Another domain adaptation approach for transferring information from a prior source disaster to a target disaster is to utilize unlabeled target disaster data together with labeled source data to train classifiers for the target disaster. This is possible, as information about the on-going event spreads quickly and thus unlabeled data accumulates rapidly, and can be extracted without much effort (Imran, Castillo, et al. 2015; Alam et al. 2018; H. Li, Guevara, et al. 2015; H. Li, D. Caragea, and C. Caragea 2017; H. Li, D. Caragea, C. Caragea, and Herndon 2018; H. Li, Sopova, et al. 2018). Among other works, H. Li, Guevara, et al. (2015) and H. Li, D. Caragea, C. Caragea, and Herndon (2018) proposed a domain adaptation approach based on the iterative expectation maximization (EM)/self-training (ST) strategies, used together with a weighted Naive Bayes classifier, to identify disaster relevant tweets. In this approach, a classifier is learned at each iteration, and used to label the target unlabeled data. Subsequently, the target unlabeled data, with labels assigned by the current classifier, are combined with the labeled source data and used to train the classifier at the next iteration.

As one of the early domain adaptation approaches that has been used to classify crisis tweets, the self-training approach has significantly improved the performances of the base supervised Naive Bayes classifiers. Given the recent and successful revisiting of self-training in natural language processing (NLP) (He et al. 2020; Z. Sun et al. 2020; Ye et al. 2020) and computer vision (Xie et al. 2020) in the context of deep learning models, our goal is to investigate whether self-training can further improve deep learning classifiers that take pre-trained embeddings as input, or classifiers based on pre-trained language models fine-tuned for classifying crisis tweets. More specifically, in this paper, we propose to combine self-training with Convolutional Neural Networks, and BERT models to build semi-supervised generalizable classifiers for disaster tweet classification. In particular, we used the standard BERT and also BERTweet (D. Q. Nguyen et al. 2020) models. BERTweet is a large-scale language model for English Tweets, which was pre-trained with the same architecture as BERT but with the training procedure from RoBERTa (Y. Liu et al. 2019). We evaluate the resulting classifiers on three datasets: CrisisLexT6 (Olteanu, Castillo, et al. 2014), CrisisLexT26 (Olteanu, Vieweg, et al. 2015), and 2CTweets (Schulz et al. 2017) using a leave-one-disaster-out strategy, to simulate a realistic scenario. Specifically, we design experiments where each disaster's data is used as unlabeled target data in one experiment, while all the other disasters together as used as labeled source data in that experiment. We find that when relatively large target unlabeled dataset are available, self-training can help improve CNN models significantly for some disasters, and slightly help improve BERT models. To summarize, the main contributions of this work are as follows:

- We experiment with the self-training domain adaptation approach, in combination with CNN and BERT models, for classifying disaster/crisis related tweets. To the best of our knowledge, we are the first to use self-training together with such transferable deep learning models in the disaster response domain.
- We show that BERT models pre-trained on Twitter data perform better than the standard BERT models pre-trained on English Wikipedia and BooksCorpus. This indicates that further pre-training BERT language models on large scale disaster/crisis data/tweets may further improve the performance of crisis tweet classifiers.
- We show that self-training can help improve the performance of the deep learning models (which already enable knowledge transfer through pre-training) when large amounts of target unlabeled data is available.

RELATED WORK

As we have discussed, significant progress has been made in classifying disaster/crisis related tweets in recent years. We have already mentioned relevant works in the introduction; here, we will emphasize domain adaptation approaches for classifying crisis tweet, as well as works that used CNN or BERT models, especially in disaster response and resilience.

In one of the early works applying deep learning to crisis tweet classification, C. Caragea, Silvescu, et al. (2016) explored the use of Convolutional Neural Networks (CNN) to classify informative tweets from six flood events in CrisisLexT26. They used three flood disaster datasets with labeled instances, tuned parameters on one dataset and then tested the resulting models on the other two test datasets. The goal was to investigate how well models with tuned parameters generalize to new events. This setting is similar to our leave-one-disaster-out setting.

D. T. Nguyen, Al-Mannai, et al. (2016) used CNNs to classify crisis related tweets, as well as situational awareness tweets. They assumed that some target labeled data is available and used two simple supervised domain adaptation techniques to combine prior source disaster data with current disaster labeled data during training. One technique was to weight the prior source disaster data, while regularizing the modified model. The other technique was to simply select a subset of the prior source disaster tweets, specifically those samples that are correctly labeled by a target-based classifier. The authors showed experimentally that CNNs that used the simple instance selection domain adaptation technique gave better results. One drawback of the abovementioned approaches is the requirement that some target labeled data is available, which is not easy to obtained especially in the beginning of a disaster.

Burel and Alani (2018) used CrisisLexT26 dataset to create three different CNN-based classifiers that enabled the identification of crisis-related documents (i.e., related vs unrelated), event types (e.g., hurricane, floods, etc.) and information categories (e.g., reports on affected individuals, donations and volunteers, etc.)

In a cross-disaster setting, with the goal of building classifiers that can generalize well to different disasters/crises, H. Li, X. Li, et al. (2018) experimented with simple domain adaptation approaches that use pre-trained embedding models to transfer knowledge. More specifically, the authors experimented with embeddings at both word-level and sentence-level, under the multi-source domain adaptation setting or cross-disaster setting, using traditional supervised learning classifiers, such as Support Vector Machines (SVM). When comparing different types of pre-trained word embeddings and crisis-specific word embeddings, they found that GloVe word embeddings pre-trained on Twitter data generally performed better for the CrisisT6, CrisisT26 and 2CTweet datasets (which are also used in this paper). Ning et al. (2019) has experimented with CNN models in a cross-disaster setting, and showed that CNN models outperformed other models on the informativeness task of the CrisisLexT26 dataset. Kersten et al. (2019) proposed a parallel CNN in a cross-disaster settings and successfully used it on different data collections for classifying disaster related data.

Related to the use of BERT models for cross-disaster classification, Wiegmann et al. (2020) experimented with BERT models on 46 disasters of 9 different types. They found that detection models worked equally well over a broad range of disaster types when being trained for the respective type. However, domain transfer across disasters of different types was shown to lead to unacceptable performance drops. Desai et al. (2020) built an emotion dataset of 15,000 English tweets spanning three hurricanes: Harvey, Irma, and Maria. They presented a comprehensive study of fine-grained emotions and proposed classification tasks to discriminate between coarse-grained emotion groups. They used unlabeled Twitter data to further pre-train the standard BERT model, which achieved the best results overall, as compared to other models, such as CNNs. Ma (2019) applied BERT for multi-class crisis tweet classification (specifically, tweets were classified according to several situational awareness categories). Ma (2019) proposed to use CNNs, as well as recurrent neural networks (RNNs) on top of the representations from BERT, to perform classification. Experimental results using several disaster datasets, including CrisisLexT26, showed that the results of the standard BERT model with one classification layer (without the addition of CNN/RNN models), i.e. fine-tuning had better recall than adding CNN or RNN on top of BERT embeddings. The experiments here were run on all the data ignore disasters difference, not in domain adaptation or cross disaster types setting.

J. Liu et al. (2020) proposed an end-to-end transformer-based model CrisisBERT for two crisis classification tasks: crisis detection and crisis recognition. They also proposed document-level contextual embedding Crisis2Vec for crisis embedding. They also used CrisisLexT6 and CrisisLexT26 datasets for crisis detection and crisis type recognition classification tasks.

In terms of domain adaptation based on self-training, H. Li, Guevara, et al. (2015) proposed a domain adaptation approach based on the iterative EM algorithm and a weighted Naive Bayes classifier, for identifying disaster relevant tweets. In this approach, a classifier is learned at each iteration, and used to label the target unlabeled data. Subsequently, the target data, with probabilistic soft-labels assigned by the current classifier (e.g., $p(+|d) = 0.7$ and

$p(-|d) = 0.3$ for an instance d), are combined with the labeled source data and used to train the classifier at the next iteration. The original classifier is trained from source data only. The process continues for a fixed number of iterations, or until convergence, by slowly giving more weight to the soft-labeled target data during training. H. Li, D. Caragea, C. Caragea, and Herndon (2018) extended this work by replacing the EM strategy with the self-training strategy. Similar to the EM domain adaptation, the self-training domain adaptation is an iterative approach that uses a weighted Naive Bayes classifier to combine source and target data. Just like the EM approach, it starts by learning a supervised classifier from source data only, and uses that classifier to label the target unlabeled data. However, instead of adding all the target data with probabilistic soft-labels to the training set for the next iteration as in EM, in self-training only the most confidently classified data are added to the training set, with hard (e.g., +/- or 1/0) labels. (H. Li, Sopova, et al. 2018) compared the self-training approach with a feature representation based domain adaptation approach, Correlation Alignment (CORAL) (B. Sun et al. 2015), used on top of supervised Naive Bayes classifiers, and proposed a hybrid model combining self-training and CORAL on top of Naive Bayes models. They experimented with different source and target disaster pairs of CrisisLexT6 on predicting the disaster related tweets. They found that for some target disaster the hybrid model performed better but overall self-training approach performance was still hard to beat

Alam et al. (2018) proposed a domain adaptation approach that combines domain adversarial training and graph embeddings with a classification network. The adversarial training is used to reduce the distribution shift, while the graph embeddings are used to induce structural similarity between source and target data. Yao and Wang (2020) proposed to apply a domain-adversarial neural network to perform sentiment analysis of tweets posted during hurricanes. Their method first retrieves hurricane-relevant tweets with a supervised trained Random Forest classifier, then classifies the sentiment of the retrieved tweets with the domain-adversarial neural network.

Finally, BERT has been used for domain adaptation in other NLP applications. For example, Du et al. (2020) investigated how to efficiently apply the pre-training language model BERT for single source unsupervised domain adaptation with application to sentiment analysis of the Amazon multi-domain data. They first used a pre-trained BERT model and further pre-trained it with two tasks: 1) domain-distinguishing task, where instead of predicting the next sentence in the original BERT model, they predicted whether two sentences are from the same domain or from different domains; 2) target domain masked language modeling (MLM), where they further pre-trained the model with MLM on the target domain to inject target domain knowledge. With the post-trained BERT model, they further proposed to use adversarial training. This combination of post-training and adversarial training was shown to be better than just fine-tuning or adversarial training with vanilla (default pre-trained) BERT model.

METHOD

Let $D_S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_{m_s}, y_{m_s})\} \subseteq \mathcal{X} \times \mathcal{Y}$ be the set of labeled examples[1] where $\mathcal{X}$ is the feature space and $\mathcal{Y}$ is the label space. Let $D_U = \{\mathbf{x}_1, \ldots, \mathbf{x}_{m_u}\} \subseteq \mathcal{X}$ be the target unlabeled data, and $D_T = \{\mathbf{x}_1, \ldots, \mathbf{x}_{m_t}\} \subseteq \mathcal{X}$ be the target test data. Combining self-training with deep learning models (base models) approach can be illustrated by the procedure show in Algorithm 1. The base models we choose in this paper are CNN model and BERT models. For CNN models, we use the architecture proposed by Kim (2014), and we use BERT default model and another variant BERTweet model (D. Q. Nguyen et al. 2020) which is pre-trained on specifically on tweets. Some details of the models are discussed in Experimental Setup section, for more details about these base models we refer the reader to the original papers.

There are different strategies that can be used here as discussed in related works, we choose to first experiment with a way similar to soft-labeling strategy in this paper for simplicity. That means, we will add all target unlabeled instances to the training set with positive and negative labels but calculate the their corresponding loss with probabilities as weights. Concretely, in the first step, the base model is trained with only D_S, and the model is trained to minimize the cross entropy loss of source labeled instances only. Then in the second step, we use this base model to make predictions on target unlabeled data and get the probability $p(y_j = k)$ for instance j in D_U to be class $k \in \mathcal{Y}$ (positive or negative). Then in the third step, we will add the pseudo-labeled D_U instances to D_{UL} along with weight λ_{uj} for the jth pseudo-labeled instance. Then we will train a new model by using both D_S and D_{UL} and minimize the weighted loss.

$$\mathcal{L} = \sum_{i=1}^{m_s} \lambda_{si} * l_{si} + \sum_{j=1}^{m_{ul}} \lambda_{uj} * l_{uj} \tag{1}$$

[1]To apply the proposed approach on multiple sources problems like in this paper, let $D_{S_i} = \{(\mathbf{x}_1^{(i)}, y_1^{(i)}), \ldots, (\mathbf{x}_{s_i}^{(i)}, y_{s_i}^{(i)})\}$ be the set of ith source domain labeled examples. We can simply combine all D_{S_i} together as $D_S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_{m_s}, y_{m_s})\} \subseteq \mathcal{X} \times \mathcal{Y}$, or use instance selection techniques to form D_S. In this paper, we use the first option, meaning simply combine all available labeled sources instances.

where l_i is the cross entropy loss of the ith source labeled instance, λ_{si} is the weight for it and it's set to 1, l_j is the cross entropy loss of the jth target pseudo-labeled data D_{UL}, λ_{uj} is the weight for it. For simplicity, we just use $p(y_j = k)$ as the weights[2]. Finally, we can iterate on the third step to a certain number, and use the final model to make predictions on the target test data.

Algorithm 1 Self-training with deep learning models

Input: Labeled instances set D_S and target unlabeled instances set D_U, target test set D_T.
Output: Predictions for target test set.

1: Initialize weights for D_S instances $\boldsymbol{\lambda_s} \leftarrow \mathbf{1}$.
2: Initialize target pseudo-labeled data $D_{UL} \leftarrow \emptyset$, and $\boldsymbol{\lambda_u} \leftarrow \mathbf{0}$ for D_{UL} instances.
3: **for** $i = 0$ to iterations **do**
4: Train the base deep learning model with D_S and D_{UL} to minimize the weighted loss in Equation 1.
5: Use the model to label instances in D_U to be all positive (+) with probabilities $\mathbf{p}_+$ and all being negative(-) with probabilities $\mathbf{p}_-$.
6: Update $D_{UL} \leftarrow D_U^+ + D_U^-$, with $\boldsymbol{\lambda_u} \leftarrow \mathbf{p}_+ + \mathbf{p}_-$.
7: **end for**
8: Use the model to make final predictions P on target test data D_T.
9: **return** P

DATASET

We use three datasets in this study, specifically: 1) CrisisLexT6 (Olteanu, Castillo, et al. 2014); 2) CrisisLexT26 (Olteanu, Vieweg, et al. 2015); and 3) 2CTweets (Schulz et al. 2017).

CrisisLexT6 is a collection of English tweets crawled during 6 disasters that occurred between October 2012 and July 2013 in USA, Canada and Australia, as shown in the first part of Table 1. There are approximately 10,000 tweets for each disaster, all manually labeled as *on-topic* (i.e., relevant) or *off-topic* (i.e., irrelevant) using the crowd-sourcing platform CrowdFlower (currently, renamed Appen). CrisisLexT26 is a collection of tweets posted during 26 crisis events that happened in 2012 or 2013. Given that we use English embeddings, we only selected 7 events from the 26 events in our study, as shown in the middle part of Table 1, and focused on the task of classifying tweets as *informative* or *non-informative*. 2CTweets is a collection of tweets about incidents, such as car crashes, fires or shootings, which happened in 10 different cities, as shown in the last part of Table 1. Tweets in 2CTweets were labeled as incident related (*Yes*) or not (*No*). Given that incidents in different cities most likely involve local named entities, such as local street names, adaptation is needed to enable generalization of classifiers between different cities (Schulz et al. 2017).

For CNN models, we first perform preprocessing for all datasets, using a Python version of the GloVe's Ruby preprocessing script, The statistics of each class in each event dataset, before and after the preprocessing, together with the total number of tweets in the dataset, are shown in Table 1 for the three datasets, respectively.

One of the BERT variants used, BERTweet (D. Q. Nguyen et al. 2020), has its own tokenizer. Thus, in the case of BERTweet, the preprocessing can be done by this tokenizer, which is recommended as special tweet tokens are included in the pre-trained BERTweet. One example of preprocessing performed by the BERTweet tokenizer is to convert user mentions and web/url links into special tokens of the form @USER and HTTPURL, respectively. We feed the BERTweet models with the raw tweets corresponding to the cleaned tweets used in the CNN models. To be consistent, we use the raw tweets for the standard BERT model too. Although standard BERT has its own tokenizer, this tokenizer is not designed specifically for tweets.

EXPERIMENTAL SETUP

We evaluate the models using a leave-one-out and 5-folds cross-validation (CV) setting, as described below.

Leave-one-out: We use a leave-one-out setting for evaluation to simulate a real scenario. Namely, for each dataset (e.g., CrisisLexT6), in a particular experiment, we select one event as the target test data, and use the rest of the events from that dataset as source training data. For example, when Hurricane Sandy from CrisisLexT6 is selected as test, the other five disasters from CrisisLexT6 are used for training. Thus, we combine all sources into one training set, the simplest strategy for multi-source domain adaptation. Each disaster is left out in one experiment.

[2]From optimization perspective, this can be definitely improved. The intuition is that the probabilities can sever as the confidence level of the pseudo-labeled instances being positive and negative. By adding these weights, the weighted loss will take consideration of pseudo-labeled instances but still be dominated by the losses of source labeled instances.

Table 1. Statistics about the datasets (CrisisLexT6, CrisisLexT26, and 2CTweets), before and after cleaning

	Before Cleaning			After Cleaning		
CrisisLexT6	On-topic	Off-topic	Total	On-topic	Off-topic	Total
2012_Sandy_Hurricane	6138	3870	10008	5443	3757	9200
2013_Queensland_Floods	5414	4619	10033	3324	4530	7854
2013_Boston_Bombings	5648	4364	10012	4824	4301	9125
2013_West_Texas_Explosion	5246	4760	10006	4123	4711	8843
2013_Oklahoma_Tornado	4827	5165	9992	4101	5111	9212
2013_Alberta_Floods	5189	4842	10031	4550	4745	9295
CrisisLexT26	Informative	Non-Inf.	Total	Informative	Non-Inf.	Total
2012_Colorado_wildfires	685	268	953	665	252	917
2013_Queensland_floods	728	191	919	681	183	864
2013_Boston_bombings	417	512	929	397	489	886
2013_West_Texas_explosion	472	439	911	444	390	834
2013_Alberta_floods	685	298	983	665	284	949
2013_Colorado_floods	768	157	925	736	147	883
2013_NY_train_crash	904	95	999	684	88	772
2CTweets	Yes	No	Total	Yes	No	Total
Memphis	361	721	1082	333	699	1032
Seattle	800	1404	2204	739	1293	2032
NYC	413	1446	1859	373	1411	1784
Chicago	214	1270	1484	202	1254	1456
San Francisco	304	1176	1480	290	1146	1436
Boston	604	2216	2820	586	2123	2709
Brisbane	689	1898	2587	667	1746	2413
Dublin	199	2616	2815	189	2574	2763
London	552	2444	2996	490	2287	2777
Sydney	852	1991	2843	832	1947	2779

5-folds-CV: For a thorough evaluation of the models with self-training, we split the target data into 5 folds; we use 1 fold (20%) as target test data, and the remaining 4 folds (80%) as target unlabeled data, and run 5-fold cross-validation to evaluate the performance.

Some details of the models are as follows:

- CNN: For CNN models, we use the architecture proposed by Kim (2014). The filter sizes are 3, 4, 5, with 100 feature maps each. A dropout rate of 0.5 is used for regularization purposes. Furthermore, ReLu is used as the activation function of the hidden layers, and the sigmoid function for final output layer. Based on results from previous work (H. Li, X. Li, et al. 2018), for the CNN models, we use GloVe embeddings pre-trained on Twitter data for CrisisLexT6 and CrisisLexT26, and Word2Vec for 2CTweets data. The pre-trained word embeddings are loaded as the embedding layer. We use 80% of all source labeled data for training and 20% for validation and apply early stopping with two steps[3]. The mini-batch size during training is 128 for all three datasets, optimizer is Adam with learning rate 0.001, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 1e{-}7$.

- CNN-ST: CNN-ST models are CNN models combined with self-training. For each dataset, the CNN-ST model hyper-parameters are set to be the same as those of the corresponding CNN models. For all experiments, we run the self-training approach for three iterations[4].

- BERT: This is the standard BERT language model, specifically, "bert-base-uncased", with a classification layer added for sequence classification. This model uses the average of final hidden states of all tokens (all words in a tweet), an averaged 768 dimension vector, as the aggregation of a tweet[5]. The whole model is fine-tuned with all source labeled instances. For CrisisLexT6, the mini-batch size is 128, and the maximum

[3]Given the there multiple source domains and the data distribution disparate exists, we found that using only 80% source labeled data and validate on the 20% then test on target data gives better results than use all available source instances from all source domains

[4]Considering that CNNs are slow and expensive to train compared with Naive Bayes, for example, we only experiment with small iteration numbers. Specifically, the results reported here are for three iterations. Running more iterations - up to 10 - didn't show a significant improvement as compared to just three iterations.

[5]We have experimented with this representation final hidden vector of the first input token [cls] from the BERT language model (a 768 dimension vector) as the embedding of a tweet. The results are not much different with CrisisLexT6 and 2CTweets, but slightly worse for CrisisLexT26.

sequence length is 80. For CrisisLexT26, the mini-batch size is 32 and the maximum sequence length is 64. For 2CTweet, the mini-batch size is 64 and the maximum sequence length is 64.

- BERTweet: This model has just one difference from the standard BERT: The base language model is BERTweet, and BERTweet tokenizer is used. The whole model is then fine-tuned with all source labeled instances.
- BERTweet-ST: BERTweet-ST is the BERTweet model together with self-training. As BERTweet models are even more expensive to train than CNN models, we only add the target unlabeled data once, i.e. run self-training with one iteration only.

For all BERT related models, Adam with weight decay is used for optimization, learning rate 1e−5, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 1e{-}8$.

RESULTS AND DISCUSSION

We report the accuracy and macro F1 score of each model trained. Specifically, the results of the experiments for CrisisLexT6 are shown in Table 2, the results for CrisisLexT26 are shown in Table 3 and the results for 2CTweets are shown in Table 4. We will discuss the results with respect to several research questions in what follows based on mainly comparing macro F1 results as the trends are consistent with accuracy results.

How do the CNN, BERT, BERTweet models perform across all datasets?

We first compare how the models perform without self-training. From the results we can see that the BERTweet models have the best overall performance for all three datasets, and especially for CrisisLexT6, where the BERTweet model is better than the CNN model for every target disaster. For CrisisLexT26, the performance of the BERTweet model is better than that of the CNN model for different target disasters except Colorado Wildfires where the results can also be seen as equivalent. For 2CTweets, the performance of the BERTweet model is better than that of the CNN model except for San Francisco where CNN model is better. For CrisisLexT6 and CrisisLexT26, BERTweet model is better than BERT model for almost all target disasters (except Queensland Floods in CrisisLexT6, Colorado Wildfires and Colorado Floods in CrisisLexT26). Although, the differences between these two models are not that significant for 2CTweets, we can still see better results with BERTweet model for some target for example Sydney. In general, because BERTweet is specifically trained on tweets, and tweet special tokens, such as user mentions, are included in BERTweet, but not in the standard BERT model, we can do fine-tuning the BERTweet language model for crisis tweets analysis. This also suggests that a language model that is specifically trained on large scale disaster/crises data may further help the downstream crisis data classification tasks.

The results also show that the BERT default models are generally better than the CNN models, although for CrisisLexT6, the CNN model has performance similar to that of the BERT model. Given that the BERTweet models are better than the BERT models, we only experiment with self-training with BERTweet models.

Can self-training be used to further improve the performance of the CNN and BERTwet models?

To answer this question, we will compare the results of the CNN and CNN-ST models (top two rows of the result tables of both accuracy and Macro F1 scores), and also the results of the BERTweet and BERTweet-ST models (last two rows in the result tables of both accuracy and Macro F1 scores). From Table 2, we can see that, for CrisisLexT6, self-training helps improve the base CNN models, and slightly helps improve the BERTweet base model. CNN-ST improves the performances of CNN base model for three out of six target disasters(Sandy Hurricane, Boston Bombing and West Texas Explosion), slightly improves the results for one disaster (Oklahoma Tornado), and achieve equivalent results as CNN base model for two target disasters (Queensland Floods and Alberta Floods). We can observe similar pattern when comparing BERTweet-ST with the BERTweet base model, except that for West Texas Explosion, BERTweet base model already performs very good and adding self-training doesn't help. Furthermore, for all three datasets, BERTweet-ST models are better than the CNN-ST models, as the corresponding base models are also better (a result consistent with prior works). A closer look into the accuracy and macro F1 score results of each disaster in Table 2 show that the overall improvements of the CNN-ST and BERTweet-ST models, as compared to their corresponding base models, are mainly due to the better performance on Hurricane Sandy, and Boston Bombing. Intuitively, the improvements on these events, when using self-training, make sense as these are unique disasters of their type in the CrisisLexT6 collection. Therefore, adding back the unlabeled data from those respective disasters, e.g., Hurricane Sandy, helps the base model learn representations specific to Hurricane Sandy. Although Boston Bombing and West Texas Explosion have been shown before to be close in data distributions (H. Li, D. Caragea, C. Caragea, and Herndon 2018), they are still different types of man-made

Table 2. Accuracy and macro F1 results of each disaster in CrisisLexT6 in leave-one-out setting. The best averaged result is highlighted in bold.

CrisisLexT6 Accuracy	Sandy Hurricane	Queensland Floods	Boston Bombing	West Texas Explosion	Oklahoma Tornado	Alberta Floods	Accuracy Average
CNN	84.52	94.63	86.02	93.71	91.68	88.74	89.88
CNN-ST	91.42	95.11	91.27	95.31	92.21	88.08	92.23
BERT	86.98	96.24	85.35	95.04	92.40	86.99	90.50
BERTweet	88.25	95.81	90.54	97.39	94.04	95.99	93.67
BERTweet-ST	91.35	95.93	92.89	97.51	94.53	96.02	**94.70**
Macro F1	Sandy Hurricane	Queensland Floods	Boston Bombing	West Texas Explosion	Oklahoma Tornado	Alberta Floods	Macro F1 Average
CNN	84.43	94.52	86.00	93.65	91.55	88.65	89.80
CNN-ST	91.20	94.99	91.18	95.27	92.17	88.07	92.15
BERT	86.84	96.17	85.32	94.99	92.29	86.84	90.41
BERTweet	88.09	95.73	90.54	97.37	93.97	95.99	93.62
BERTweet-ST	91.12	95.85	93.22	97.58	94.48	96.02	**94.71**

Table 3. Accuracy and macro F1 results of each disaster in CrisisLexT26 in leave-one-out setting. The best averaged result is highlighted in bold.

CrisisLexT26 Accuracy	Colorado Wildfires	Queensland Floods	Boston Bombings	West Texas Explosion	Alberta Floods	Colorado Floods	NY Train Crash	Accuracy Average
CNN	85.93	86.11	84.20	87.53	81.45	87.20	94.30	86.67
CNN-ST	86.37	86.92	84.31	86.09	78.72	88.44	94.69	86.51
BERT	87.35	86.81	86.79	88.73	82.19	89.69	95.85	88.20
BERTweet	86.04	87.15	87.47	91.49	83.88	89.58	96.89	**88.93**
BERTweet-ST	86.80	87.96	87.92	91.13	83.14	89.24	96.89	**89.01**
Macro F1	Colorado Wildfires	Queensland Floods	Boston Bombings	West Texas Explosion	Alberta Floods	Colorado Floods	NY Train Crash	Macro F1 Average
CNN	82.75	79.20	84.14	87.40	76.69	76.09	83.76	81.43
CNN-ST	81.71	79.57	84.26	85.92	70.38	76.33	85.09	80.47
BERT	83.59	80.70	86.75	88.70	77.59	81.18	88.59	83.87
BERTweet	82.62	81.91	87.32	91.46	78.76	80.25	91.34	**84.81**
BERTweet-ST	83.46	82.72	87.79	91.10	78.58	78.52	91.20	**84.77**

Table 4. Accuracy and macro F1 score results of each disaster in 2CTweets in leave-one-out setting. The best averaged result is highlighted in bold.

2CTweets Accuracy	Memphis	Seattle	NYC	Chicago	San Francisco	Boston	Brisbane	Dublin	London	Sydney	Accuracy Average
CNN	87.79	84.01	92.32	94.37	92.90	94.28	91.05	97.32	93.59	94.24	92.19
CNN-ST	89.15	85.78	92.43	94.44	91.85	94.72	91.17	97.39	93.73	94.53	92.52
BERT	89.83	88.88	93.33	95.12	91.85	96.20	91.59	97.79	95.35	94.57	93.45
BERTweet	90.79	89.42	94.11	95.47	90.74	96.68	91.79	97.72	95.61	97.41	**93.97**
BERTweet-ST	91.09	88.63	93.89	95.74	90.39	96.46	91.13	97.76	95.57	97.59	**93.82**
Macro F1	Memphis	Seattle	NYC	Chicago	San Francisco	Boston	Brisbane	Dublin	London	Sydney	Macro F1 Average
CNN	85.50	82.32	88.02	88.64	89.23	91.02	88.88	89.17	87.87	92.90	88.36
CNN-ST	87.38	84.62	88.06	88.43	87.85	91.89	89.29	90.04	88.02	93.23	88.88
BERT	88.58	88.24	90.12	90.33	88.33	94.43	89.88	91.91	91.53	93.32	90.67
BERTweet	89.81	88.92	91.33	90.89	87.15	95.17	90.11	91.87	92.01	96.87	**91.41**
BERTweet-ST	90.10	88.15	91.08	91.28	86.76	94.87	89.45	91.99	92.00	97.08	**91.28**

disasters, and thus adding back the target unlabeled data through self-training helps improve the performance, especially that of the base CNN model.

However, self-training doesn't help much for 2CTweets and CrisisLexT26. The reason may be that we do not have enough target unlabeled data in CrisisLexT26 and 2CTweets, as compared to CrisisLexT6 for which larger amounts of unlabeled data are available in the target event. For reference, the size of each dataset is shown in Table 1. As can be seen, for CrisisLexT6, for each target disaster, we have nearly 8,000 tweets, therefore about 6,400 target unlabeled tweets. However, for CrisisLexT26, each target disaster has only about 1,000 tweets in total, and some disaster datasets are highly unbalanced (e.g., in New York train crash case). For 2CTweets, the size of each target disaster/crisis is more than 1,000, but still less than 3,000. With our leave-one-out setting, the number of source instances is roughly 10 times the number of target unlabeled instances, and therefore, the source instances still dominate the models' performances. Our conclusion is that self-training should, in general, help deep learning models when a very large number of target unlabeled data is available. However, there is a trade-off, as the training cost increases significantly as compared to the case when each model is trained only once (and this cost may not be justified, especially in the early hours of a disaster). Furthermore, the training cost is even higher for a larger dataset. In our experiments, one positive outcome was that the results obtained with 10 iterations were comparable to those with just 3 iterations for the CNN-ST models (results now shown).

CONCLUSIONS

In this paper, we propose to combine self-training with CNN and BERT models to improve the performance of tweet classifiers for a target disaster, where labeled data is assumed to be unavailable, while unlabeled data is readily available. Concretely, we first run CNN classifiers, BERT based classifiers, and BERTweet based classifiers on three crisis tweet collections, CrisisLexT6, CrisisLexT26 and 2CTweets. We find that the language model BERTweet, which is the BERT language model further pre-trained on tweets, is better than the standard BERT when fine-tuned for downstream tweets classification tasks. We then combine self-training with CNN classifiers and BERTweet fine-tuned classifiers, and compare the results of the resulting self-trained models with those of the base models on the three datasets considered in the study. We find that self-training can help improve the CNN and BERTweet models for CrisisLexT6, but not for CrisisLexT26 and 2CTweets, where data sizes are smaller than those in CrisisLexT6. Therefore, we conclude that self-training can help when a large amount of target unlabeled data is available.

To further verify this, in future work, we plan to run the models on large collections of crisis tweets, including collections from CrisisNLP[6] or TREC Incident Streams[7]. As these collections contain tweets that are labeled with situational awareness categories or information types, we will experiment with the self-training approach on multi-class classification tasks, which are more challenging than the binary classification tasks used in this paper.

In this paper, we used target unlabeled data for self-training with pre-trained language models. However, another way to use target unlabeled data is to further pre-train the language models, for example BERTweet, with the target unlabeled data, especially when the size of the target unlabeled data is very large. Following the pre-training, we can fine-tune the models with a small amount of target labeled data, and build a classifier for the target disaster. Furthermore, as we have seen benefits from the BERTweet as compared with the standard BERT, we plan to train a BERT-Crisis language model on a large amount of crisis data. We can also specifically train BERTweet-Crisis just for Twitter data since tweets have unique characteristics (e.g. short, special user mentions, hashtags). We also plan to use other self-training strategies and domain adaptation techniques, such as hard labeling instead of soft-labeling, and instance selection or weighting in combination with self-training for BERTweet models. Given that fine-tuning a language model, such as BERTweet is easy to over-fit, it's better that we select the closest source instances to the target disaster domain for fine-tuning. Instance selection with help from reinforcement learning such as proposed by Ye et al. (2020) is also interesting. Finally, self-training approaches that can handle multi-modal social media data are also of interest. We plan to explore multi-modal models to build better generalized classifiers across different types of disasters/crises.

REFERENCES

Alam, F., Joty, S. R., and Imran, M. (2018). "Domain Adaptation with Adversarial Training and Graph Embeddings". In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*. Ed. by I. Gurevych and Y. Miyao. Association for Computational Linguistics, pp. 1077–1087.

[6]https://crisisnlp.qcri.org/

[7]http://dcs.gla.ac.uk/ richardm/TREC_IS/

Burel, G. and Alani, H. (2018). "Crisis Event Extraction Service (CREES) - Automatic Detection and Classification of Crisis-related Content on Social Media". In: *15th International Conference on Information Systems for Crisis Response and Management*.

Caragea, C., Silvescu, A., and Tapia, A. H. (2016). "Identifying Informative Messages in Disasters using Convolutional Neural Networks". In: *13th Proceedings of the International Conference on Information Systems for Crisis Response and Management, Rio de Janeiro, Brasil, May 22-25, 2016*. Ed. by A. H. Tapia, P. Antunes, V. A. Bañuls, K. A. Moore, and J. P. de Albuquerque. ISCRAM Association.

Caragea, C., Squicciarini, A. C., Stehle, S., Neppalli, K., and Tapia, A. H. (2014). "Mapping moods: Geo-mapped sentiment analysis during hurricane sandy". In: *11th Proceedings of the International Conference on Information Systems for Crisis Response and Management, University Park, Pennsylvania, USA, May 18-21, 2014*. Ed. by S. R. Hiltz, L. Plotnick, M. Pfaf, and P. C. Shih. ISCRAM Association.

Castillo, C. (2016). *Big Crisis Data: Social Media in Disasters and Time-Critical Situations*. Cambridge University Press.

Coche, J., Montarnal, A., Tapia, A., and Benaben, F. (May 2020). "Automatic Information Retrieval from Tweets: A Semantic Clustering Approach". In: *ISCRAM 2020 - 17th International conference on Information Systems for Crisis Response and Management*. Balcksburg, United States, p.134–141.

Desai, S., Caragea, C., and Li, J. J. (2020). "Detecting Perceived Emotions in Hurricane Disasters". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*. Ed. by D. Jurafsky, J. Chai, N. Schluter, and J. R. Tetreault. Association for Computational Linguistics, pp. 5290–5305.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *CoRR* abs/1810.04805. arXiv: 1810.04805.

Du, C., Sun, H., Wang, J., Qi, Q., and Liao, J. (2020). "Adversarial and Domain-Aware BERT for Cross-Domain Sentiment Analysis". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*. Ed. by D. Jurafsky, J. Chai, N. Schluter, and J. R. Tetreault. Association for Computational Linguistics, pp. 4019–4028.

Ghafarian, S. H. and Yazdi, H. S. (2020). "Identifying crisis-related informative tweets using learning on distributions". In: *Inf. Process. Manag.* 57.2, p. 102145.

Grace, R., Halse, S. E., Kropczynski, J., Tapia, A. H., and Fonseca, F. (2019). "Integrating Social Media in Emergency Dispatch via Distributed Sensemaking". In: *Proceedings of the 16th International Conference on Information Systems for Crisis Response and Management, València, Spain, May 19-22, 2019*. Ed. by Z. Franco, J. J. González, and J. H. Canós. ISCRAM Association.

He, J., Gu, J., Shen, J., and Ranzato, M. (2020). "Revisiting Self-Training for Neural Sequence Generation". In: *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Hiltz, S. R., Hughes, A. L., Imran, M., Plotnick, L., Power, R., and Turoff, M. (2020). "Exploring the usefulness and feasibility of software requirements for social media use in emergency management". In: *International Journal of Disaster Risk Reduction* 42, p. 101367.

Homeland Security (2014). *Using Social Media for Enhanced Situational Awareness and Decision Support*. Virtual Social Media Working Group and DHS First Responders Group.

Imran, M., Castillo, C., Diaz, F., and Vieweg, S. (2015). "Processing Social Media Messages in Mass Emergency: A Survey". In: *ACM Comput. Surv.* 47.4, 67:1–67:38.

Imran, M., Elbassuoni, S., Castillo, C., Diaz, F., and Meier, P. (2013). "Practical extraction of disaster-relevant information from social media". In: *22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013, Companion Volume*. Ed. by L. Carr, A. H. F. Laender, B. F. Lóscio, I. King, M. Fontoura, D. Vrandecic, L. Aroyo, J. P. M. de Oliveira, F. Lima, and E. Wilde. International World Wide Web Conferences Steering Committee / ACM, pp. 1021–1024.

Imran, M., Ofli, F., Caragea, D., and Torralba, A. (2020). "Using AI and Social Media Multimodal Content for Disaster Response and Management: Opportunities, Challenges, and Future Directions". In: *Inf. Process. Manag.* 57.5, p. 102261.

Kaufhold, M.-A., Bayer, M., and Reuter, C. (2020). "Rapid relevance classification of social media posts in disasters and emergencies: A system and evaluation featuring active, incremental and online learning". In: *Inf. Process. Manag.* 57.1.

Kersten, J., Kruspe, A. M., Wiegmann, M., and Klan, F. (2019). "Robust filtering of crisis-related tweets". In: *Proceedings of the 16th International Conference on Information Systems for Crisis Response and Management, València, Spain, May 19-22, 2019*. Ed. by Z. Franco, J. J. González, and J. H. Canós. ISCRAM Association.

Kim, Y. (2014). "Convolutional Neural Networks for Sentence Classification". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*. Ed. by A. Moschitti, B. Pang, and W. Daelemans. ACL, pp. 1746–1751.

Li, H., Caragea, D., and Caragea, C. (2017). "Towards Practical Usage of a Domain Adaptation Algorithm in the Early Hours of a Disaster". In: *14th Proceedings of the International Conference on Information Systems for Crisis Response and Management, Albi, France, May 21-24, 2017*. Ed. by T. Comes, F. Bénaben, C. Hanachi, M. Lauras, and A. Montarnal. ISCRAM Association.

Li, H., Caragea, D., Caragea, C., and Herndon, N. (2018). "Disaster response aided by tweet classification with a domain adaptation approach". In: *Journal of Contingencies and Crisis Management* 26.1, pp. 16–27. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/1468-5973.12194.

Li, H., Guevara, N., Herndon, N., Caragea, D., Neppalli, K., Caragea, C., Squicciarini, A. C., and Tapia, A. H. (2015). "Twitter Mining for Disaster Response: A Domain Adaptation Approach". In: *12th Proceedings of the International Conference on Information Systems for Crisis Response and Management, Krystiansand, Norway, May 24-27, 2015*. Ed. by L. Palen, M. Büscher, T. Comes, and A. L. Hughes. ISCRAM Association.

Li, H., Li, X., Caragea, D., and Caragea, C. (2018). "Comparison of Word Embeddings and Sentence Encodings as Generalized Representations for Crisis Tweet Classification Tasks". In: *Proceedings of ISCRAM Asia Pacific 2018: Innovating for Resilience – 1st International Conference on Information Systems for Crisis Response and Management Asia Pacific*. Ed. by K. Stock and D. Bunker, pp. 480–493.

Li, H., Sopova, O., Caragea, D., and Caragea, C. (2018). "Domain Adaptation for Crisis Data Using Correlation Alignment and Self-Training". In: *Int. J. Inf. Syst. Crisis Response Manag.* 10.4, pp. 1–20.

Liu, J., Singhal, T., Blessing, L. T. M., Wood, K. L., and Lim, K. H. (2020). "CrisisBERT: a Robust Transformer for Crisis Classification and Contextual Crisis Embedding". In: *CoRR* abs/2005.06627. arXiv: 2005.06627.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). "RoBERTa: A Robustly Optimized BERT Pretraining Approach". In: *CoRR* abs/1907.11692. arXiv: 1907.11692.

Ma, G. (2019). "Tweets Classification with BERT in the Field of Disaster Management". In:

National Research Council (2013). *Public Response to Alerts and Warnings Using Social Media: Report of a Workshop on Current Knowledge and Research Gaps*. Washington, DC: The National Academies Press.

Nguyen, D. Q., Vu, T., and Nguyen, A. T. (2020). *BERTweet: A pre-trained language model for English Tweets*. arXiv: 2005.10200 [cs.CL].

Nguyen, D. T., Joty, S. R., Imran, M., Sajjad, H., and Mitra, P. (2016). "Applications of Online Deep Learning for Crisis Response Using Social Media Information". In: *CoRR* abs/1610.01030.

Nguyen, D. T., Al-Mannai, K., Joty, S. R., Sajjad, H., Imran, M., and Mitra, P. (2016). "Rapid Classification of Crisis-Related Data on Social Networks using Convolutional Neural Networks". In: *CoRR* abs/1608.03902.

Ning, X., Yao, L., Benatallah, B., Zhang, Y., Sheng, Q. Z., and Kanhere, S. S. (2019). "Source-Aware Crisis-Relevant Tweet Identification and Key Information Summarization". In: *ACM Trans. Internet Techn.* 19.3, 37:1–37:20.

Olteanu, A., Castillo, C., Diaz, F., and Vieweg, S. (2014). "CrisisLex: A Lexicon for Collecting and Filtering Microblogged Communications in Crises". In: *Proceedings of the Eighth International Conference on Weblogs and Social Media, ICWSM 2014, Ann Arbor, Michigan, USA, June 1-4, 2014*. Ed. by E. Adar, P. Resnick, M. D. Choudhury, B. Hogan, and A. H. Oh. The AAAI Press.

Olteanu, A., Vieweg, S., and Castillo, C. (2015). "What to Expect When the Unexpected Happens: Social Media Communications Across Crises". In: *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*. CSCW '15. Vancouver, BC, Canada: ACM, pp. 994–1009.

Palen, L. and Anderson, K. M. (2016). "Crisis informatics-New data for extraordinary times". In: *Science* 353.6296, pp. 224–225.

Palen, L. and Hughes, A. L. (2018). "Social Media in Disaster Communication". In: *Handbook of Disaster Research*. Ed. by H. Rodríguez, W. Donner, and J. E. Trainor. Cham: Springer International Publishing, pp. 497–518.

Plotnick, L., Hiltz, S. R., Kushma, J. A., and Tapia, A. H. (2015). "Red Tape: Attitudes and Issues Related to Use of Social Media by U.S. County-Level Emergency Managers". In: *12th Proceedings of the International Conference on Information Systems for Crisis Response and Management, Krystiansand, Norway, May 24-27, 2015*. Ed. by L. Palen, M. Büscher, T. Comes, and A. L. Hughes. ISCRAM Association.

Reuter, C., Hughes, A. L., and Kaufhold, M.-A. (2018). "Social Media in Crisis Management: An Evaluation and Analysis of Crisis Informatics Research". In: *International Journal of Human–Computer Interaction* 34.4, pp. 280–294. eprint: https://doi.org/10.1080/10447318.2018.1427832.

Reuter, C., Ludwig, T., Friberg, T., Pratzler-Wanczura, S., and Gizikis, A. (2015). "Social Media and Emergency Services?: Interview Study on Current and Potential Use in 7 European Countries". In: *IJISCRAM* 7.2, pp. 36–58.

Schulz, A., Guckelsberger, C., and Janssen, F. (2017). "Semantic Abstraction for generalization of tweet classification: An evaluation of incident-related tweets". In: *Semantic Web* 8.3, pp. 353–372.

Starbird, K., Palen, L., Hughes, A. L., and Vieweg, S. (2010). "Chatter on the red: what hazards threat reveals about the social life of microblogged information". In: *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work, CSCW 2010, Savannah, Georgia, USA, February 6-10, 2010*, pp. 241–250.

Sun, B., Feng, J., and Saenko, K. (2015). "Return of Frustratingly Easy Domain Adaptation". In: *ArXiv e-prints*. arXiv: 1511.05547 [cs.CV].

Sun, Z., Fan, C., Sun, X., Meng, Y., Wu, F., and Li, J. (2020). "Neural Semi-supervised Learning for Text Classification Under Large-Scale Pretraining". In: *CoRR* abs/2011.08626. arXiv: 2011.08626.

Verma, S., Vieweg, S., Corvey, W. J., Palen, L., Martin, J. H., Palmer, M., Schram, A., and Anderson, K. M. (2011). "Natural Language Processing to the Rescue? Extracting "Situational Awareness" Tweets During Mass Emergency". In: *Proceedings of the Fifth International Conference on Weblogs and Social Media, Barcelona, Catalonia, Spain, July 17-21, 2011*. Ed. by L. A. Adamic, R. Baeza-Yates, and S. Counts. The AAAI Press.

Wiegmann, M., Kersten, J., Klan, F., Potthast, M., and Stein, B. (2020). "Analysis of Detection Models for Disaster-Related Tweets". In: *Proceedings of the 17th International Conference on Information Systems for Crisis Response and Management, – Blacksburg, VA, USA May 2020*. ISCRAM Association.

Xie, Q., Luong, M.-T., Hovy, E. H., and Le, Q. V. (2020). "Self-Training With Noisy Student Improves ImageNet Classification". In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*. IEEE, pp. 10684–10695.

Yao, F. and Wang, Y. (2020). "Domain-specific sentiment analysis for tweets during hurricanes (DSSA-H): A domain-adversarial neural-network-based approach". In: *Comput. Environ. Urban Syst.* 83, p. 101522.

Ye, Z., Geng, Y., Chen, J., Chen, J., Xu, X., Zheng, S., Wang, F., Zhang, J., and Chen, H. (2020). "Zero-shot Text Classification via Reinforced Self-training". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*. Ed. by D. Jurafsky, J. Chai, N. Schluter, and J. R. Tetreault. Association for Computational Linguistics, pp. 3014–3024.