

Domain Adaptation with Reconstruction for Disaster Tweet Classification

Xukun Li
Kansas State University
Manhattan, Kansas
xukun@ksu.edu

Doina Caragea*
Kansas State University
Manhattan, Kansas
dcaragea@ksu.edu

ABSTRACT

Identifying critical information in real time in the beginning of a disaster is a challenging but important task. This task has been recently addressed using domain adaptation approaches, which eliminate the need for target labeled data, and can thus accelerate the process of identifying useful information. We propose to investigate the effectiveness of the Domain Reconstruction Classification Network (DRCN) approach on disaster tweets. DRCN adapts information from target data by reconstructing it with an autoencoder. Experimental results using a sequence-to-sequence autoencoder show that the DRCN approach can improve the performance of both supervised and domain adaptation baseline models.

CCS CONCEPTS

• **Computing methodologies** → **Neural networks.**

KEYWORDS

Tweet classification, deep learning, domain adaptation

ACM Reference Format:

Xukun Li and Doina Caragea. 2020. Domain Adaptation with Reconstruction for Disaster Tweet Classification. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20)*, July 25–30, 2020, Virtual Event, China. ACM, Virtual Event, China, 4 pages. <https://doi.org/10.1145/3397271.3401242>

1 INTRODUCTION

In recent years, social media platforms, such as Twitter or Facebook, have become effective communication tools, complementing more traditional communication networks (e.g., 9-1-1 centers), which are not always well-equipped to handle the large volume of requests posted by disaster-affected individuals [15]. Early works in disaster management have discussed the usefulness of social media data in improving disaster resilience, situational awareness, and emergency response [8]. To facilitate the use of social media data posted by eyewitnesses of disasters, machine learning approaches have been developed to filter informative tweets. Many of the existing studies have employed supervised learning to train a classifier on data from a disaster of interest, and used the classifier to assign labels to new

data from that same disaster. In supervised learning, data need to be manually labeled before a classifier can be trained. The data labeling process is usually expensive and time-consuming, making it hard to use supervised learning in the beginning of a disaster. However, unlabeled data from a target disaster accumulates quickly. In addition, labeled data from a prior source disaster is readily available. A supervised classifier trained on the source disaster may not perform well on the target disaster due to differences in the distributions of the two disasters (a.k.a., *covariate shift*).

Domain adaptation approaches that use labeled source disaster data, together with unlabeled target disaster data, to train a classifier on source and adapt it to the target, represent a promising solution to the filtering of information posted on social media during disasters. While domain adaptation eliminates the need for labeled target data, and has produced promising results on this task [1, 9–12], its potential is largely unexplored.

A challenge for domain adaptation is how to effectively use the information from unlabeled target data to adapt a classifier trained on labeled source data [13]. One idea is to reduce the shift between the source and target data distributions. Domain adversarial neural networks (DANN) [4] implement this idea by co-training a feature extraction network together with a source classification network. The feature extraction network aims to transform the source and target features to make them indistinguishable. Given domain invariant features, the source classification network trained on labeled source data can be effectively used to classify target data. Another way to reduce the covariate shift is to utilize domain reconstruction-classification networks (DRCN). The reconstruction network consists of an autoencoder (i.e., encoder-decoder) that reconstructs the target data in a self-supervised manner [5]. The autoencoder network is co-trained with a source classification network that shares the encoder component of the autoencoder. Being shared between the two networks, the encoder will learn information from both source and target data. As a consequence, the source network, trained on labeled source data, can be used to classify target data.

In the context of disaster tweet classification, Alam et al. [1] proposed a domain adaptation approach which combines domain adversarial training and graph embeddings with a classification network. Li et al. [11] proposed an approach based on a domain adversarial neural network, with a large convolutional neural network (CNN) as its backbone, to identify disaster images that show damage. However, there is no study on the use of domain reconstruction-classification networks on social media disaster data.

To fill in this gap, in Section 3, we describe a domain adaptation approach for tweet classification, which implements the domain reconstruction-classification idea using a sequence-to-sequence autoencoder [14]. Both the encoder and the decoder components of

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
SIGIR '20, July 25–30, 2020, Virtual Event, China

© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-8016-4/20/07...\$15.00
<https://doi.org/10.1145/3397271.3401242>

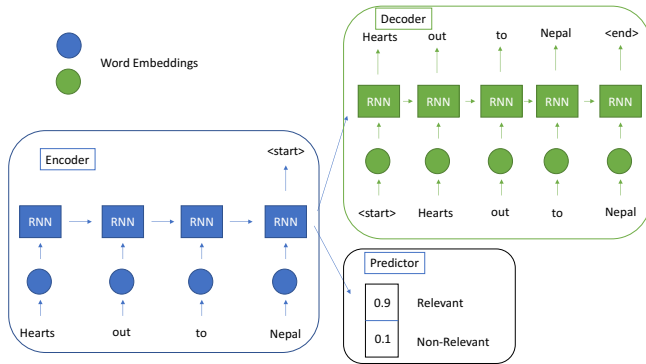


Figure 1: Architecture of the proposed domain adaptation with reconstruction network.

the autoencoder represent recurrent neural networks (RNN), specifically long short-term memory (LSTM) networks. Our experimental results in Section 4 show that the DRCN approach outperforms supervised learning approaches and also the existing approach based on domain adversarial networks and graph embeddings [1]. We conclude the paper and present ideas for future work in Section 6.

2 RELATED WORK

Social media data contributed by eyewitnesses of a disaster have been shown to be very useful for disaster management and response [8]. Domain adaptation approaches have been used to handle the lack of labeled data in the beginning of a target disaster. Some of the proposed approaches are based on traditional machine learning [9, 10, 12], while others are based on deep learning [1, 11]. For example, Li et al. [9] proposed a self-training type approach, with Naïve Bayes as a base classifier. Li et al. [10] combined a feature adaptation approach with the self-training approach, while Mazloom et al. [12] proposed a hybrid feature-instance adaptation approach. Andugula and Durbha [3] applied a domain adaptation approach, which uses spatial relationship information, to identify images that show damaged buildings post-disaster. Finally, Graf et al. [6] proposed a cross-domain classifier, which can be applied across different disaster events and different disaster types. In terms of deep learning, Alam et al. [1] and Li et al. [11] used approaches inspired from the DANN approach, as described in the introduction. In this paper, we study the DRCN approach in the context of disaster tweet classification.

3 APPROACH

In this section, we describe the components of the proposed approach. The overall architecture is shown in Figure 1.

3.1 Domain Adaptation

Let $X_s = \{(X_i^s, y_i^s)_{i=1}^{n_s}\}$ be a labeled source dataset, where n_s represents the number of instances in the source dataset, X_i^s contains a sequence of words $X_i^s = \{x_j\}_{j=1}^{|X_i^s|}$ and y_i^s is the sequence label. Let $X_t = \{(X_i^t)_{i=1}^{n_t}\}$ be an unlabeled target dataset, where n_t represents

the number of instances in the target dataset, and X_t contains a sequence of words $X_i^t = \{x_j\}_{j=1}^{|X_i^t|}$. The task in domain adaptation is to learn a classifier for the target domain using the labeled source data, X^s , together with unlabeled target data X^t . The source data and target data may share some patterns but generally have different distributions. For example, the source data may consist of tweets posted during an earthquake, while the target data may consist of tweets posted during a hurricane.

3.2 Recurrent Neural Networks

Recurrent Neural Networks (RNN) have been widely used for many tasks in Natural Language Processing, as they can capture dependencies in sequence data. Among other tasks, RNNs can be used to classify text. Formally, given a sequence (x_1, x_2, \dots, x_n) , an RNN model can predict the label y using the following forward equations:

$$h_i = RNN(x_i, h_{i-1}), i \in 1, \dots, n; y = RNN(h_n) \quad (1)$$

where h_i represents the hidden state at time step i and acts as the memory of the network (h_0 is initialized with a zero vector). General RNNs suffer from the gradient vanishing and exploding problem [7], when used with long sequences. Long short-term memory (LSTM) networks avoid this problem by introducing a cell state, in addition to the hidden state, which carries information across the sequence.

3.3 Seq2seq Autoencoder

The seq2seq model was proposed by Sutskever et al. [14] in the context of machine translation. It uses an RNN network to encode an original sequence into a hidden vector and then uses another RNN network to decode the hidden vector into another output sequence. If the original sequence and the output sequence are from different languages, the task achieved is translation. If the original sequence and the output sequence are the same, the task corresponds to autoencoding, and the model is called an autoencoder. An autoencoder can produce a sequence representation in a self-supervised manner. The representation retains all the information needed to reconstruct the original sequence itself.

3.4 Domain Adaptation with Reconstruction

Our proposed domain adaptation with reconstruction model combines the techniques discussed above. For the source data X_s , we initialize a set of word embedding w_s corresponding to words x_s in a sequence X_i^s , and then train network to predict the label for X_i^s :

$$\hat{y}_i^s = FC(Encoder(X_i^s)) \quad (2)$$

where the *Encoder* is an RNN network which extracts the encoded representation of the sequence X_i^s , and *FC* is a fully-connected layer which predicts the label of X_i^s using the representation. For the target data X_t , we train a seq2seq autoencoder on the initialized word embeddings. Formally, for each target instance X_i^t , we have:

$$\hat{X}_i^t = Decoder(Encoder(X_i^t)) \quad (3)$$

The seq2seq model and the RNN classification network share the encoder layer. Thus, the RNN encoder layer learns a good classifier on the labeled source data while using information from the unlabeled target data. The model will be trained by optimize

$$L = \min(\mathcal{L}(y_i^s, \hat{y}_i^s) + \mathcal{L}(X_i^t, \hat{X}_i^t)) \quad (4)$$

where \mathcal{L} is cross-entropy loss. The pseudocode for the training phase of the proposed model is shown in Algorithm 1.

Algorithm 1 Model Training Using SGD

Input: Source $X_s = \{(X_i^s, y_i^s)\}_{i=1}^{n_s}$, and target $X_t = \{(X_i^t)_{i=1}^{n_t}\}$

Output: $\theta_E, \theta_D, \theta_P$ (where $\theta_E, \theta_D, \theta_P$ represent the parameters in the Encoder, Decoder and Predictor components, shown in Fig. 1).

- 1: $\theta_E, \theta_D, \theta_P \leftarrow$ random initial weights
 - 2: **for** k from 1 to *steps* **do**
 - 3: **for** each source batch $\{(X_i^s, y_i^s)\}_{i=1}^m$ **do**
 - 4: Sample target unlabeled batch $(X_j^t)_{j=1}^m$
 - 5: Update θ_E, θ_P to minimize the loss $\mathcal{L}(y_i^s, \hat{y}_i^s)$
 - 6: Update θ_E, θ_P to minimize the loss $\mathcal{L}(X_i^t, \hat{X}_i^t)$
-

4 EXPERIMENTAL SETUP

4.1 Research Questions

Our experiments are designed to answer the following questions:

- 1) How do the results of our domain adaptation with reconstruction approach compare with the results of the domain adversarial with graph embeddings approach [1]?
- 2) How do the results of our domain adaptation with reconstruction approach compare with the results of the corresponding *cross-domain* supervised RNN classification models?
- 3) How do the results of the experiments that use one disaster as source compare with the results of the experiments that use the union of *all-but-one* disaster as source?

4.2 Datasets

We used two datasets in our experiments. The first dataset was published by Alam et al. [1], who made available the *train*, *validation* and *test* split they used. The dataset contains tweets crawled during Queensland Floods (QFL) and Nepal Earthquake (NEQ). Each tweet was labeled as *relevant* or *non-relevant*. Statistics about the dataset and its train/validation/test split are shown in Table 1.

Dataset	Relevant	Non-Relevant	Train	Validation	Test
NEQ	5527	6141	7000	1166	3502
QFL	5414	4619	6019	1003	3011

Table 1: Statistics of the dataset published by Alam et al. [1]

For a source-target pair, we used the *train* split of the source data to train the RNN classification network, we used the *validation* split of the target data to train the seq2seq autoencoder, and finally the *test* split of the target data to evaluate the model.

The second dataset used (called CrisisMMD) was published by Alam et al. [2]. It contains tweets crawled during 7 disaster events. Each tweet was labeled as *informative* or *non-informative*. Statistics about the CrisisMMD dataset are shown in Table 2. For each source-target experiment, we randomly split the target data into 70% *unlabeled* and 30% *test*. We used all the labeled source data to train the RNN classification network. The target *unlabeled* data is used to train the autoencoder, while the target *test* data is used to evaluate the model. We conduct experiments where the datasets

	Dataset	Not-Informative	Informative	Total
D0	California Fire	324	1162	1486
D1	Hurricane Harvey	973	3027	4000
D2	Hurricane Irma	836	3201	4037
D3	Hurricane Maria	1490	2510	4000
D4	Iraq Iran Earthquake	85	402	487
D5	Mexico Earthquake	314	925	1239
D6	Srilanka Floods	544	287	831

Table 2: Statistics of the the CrisisMMD dataset [2]

D0, D1, D2, D3, D5 are used as source datasets, respectively. We didn't use the datasets D4 and D6 as sources, as they are relatively small and unbalanced compared to the other datasets. In addition to using one disaster as source, we also performed experiments where we used *all-but-one* disasters as source and the remaining disaster as target.

4.3 Implementation Details

In the seq2seq model, both the encoder and the decoder contain one layer of LSTM cells. The dimension of the hidden vector representation is $p = 200$. We use the Adam optimizer to train the model with a learning rate of 0.001 and a mini-batch of size $m = 64$.

5 EXPERIMENTAL RESULTS

The results of the experiments on the dataset introduced in [1] are presented in Table 3. Specifically, we show accuracy (Acc), area under the ROC curve (AUC) and F1-measure (F1). In the table, RNN represents a model trained in a supervised manner (on source), while RNN+AE represents the DRCN approach, where RNN networks are used for the encoder/decoder components of the autoencoder (AE). In addition to the results of the RNN and

Model (cross-domain)	Acc	AUC	F ₁	Acc	AUC	F ₁
source -> target	QFL->NEQ			NEQ->QFL		
GE + DA [1]	n/a	58.81	59.05	n/a	66.49	65.92
RNN (supervised)	64.19	65.42	64.18	55.83	61.36	55.17
RNN+AE	68.82	72.29	68.38	81.24	87.79	81.18

Table 3: Cross-domain results for the dataset in [1]

RNN+AE models, Table 3 also shows the results of the adaptation model (GE+DA) proposed in [1]. As can be seen, the RNN model is better than the GE+DA model by a large margin. Furthermore, the table shows that the RNN+AE model achieves better results than the RNN model alone.

The results of the experiments on the CrisisMMD dataset [2] are presented in Table 4. We show the average accuracy, AUC and F1 score over five independent runs. Each row has the same target data, while each column has the same source data (except for the last column that shows the *all-but-one* results). The table also shows in-domain results (where the models are trained on target). As can be seen from the table, the results of the RNN+AE (DRCN) model are better than the results of the RNN network in most cases (including for in-domain models). While the adaptation

Method	Acc	AUC	F1	Acc	AUC	F1	Acc	AUC	F1	Acc	AUC	F1	Acc	AUC	F1			
Pair	D0→D0			D1→D0			D2→D0			D3→D0			D5→D0			all-but-D0→D0		
RNN	75.78	70.36	72.58	69.73	61.69	50.02	73.92	64.41	57.15	71.38	68.67	62.07	64.80	60.15	54.46	72.65	68.54	72.94
RNN+AE	75.11	71.37	72.11	78.25	66.77	52.88	79.45	71.71	60.27	73.99	68.43	60.37	77.88	68.74	57.44	80.27	73.27	76.24
Pair	D0→D1			D1→D1			D2→D1			D3→D1			D5→D1			all-but-D1→D1		
RNN	68.79	62.04	53.46	81.17	82.41	78.92	77.00	76.37	62.27	72.56	75.38	66.47	58.97	62.49	54.52	77.17	78.53	76.33
RNN+AE	72.00	67.79	60.97	80.92	81.84	78.66	77.53	78.56	63.92	78.25	78.37	69.29	70.22	69.46	61.32	76.92	77.79	77.55
Pair	D0→D2			D1→D2			D2→D2			D3→D2			D5→D2			all-but-D2→D2		
RNN	70.27	65.27	57.94	76.28	71.77	61.64	81.29	76.53	78.60	71.75	72.80	63.18	53.15	58.66	49.15	77.74	71.47	76.71
RNN+AE	73.21	69.59	59.87	77.05	75.94	64.14	80.54	76.91	78.13	74.00	75.13	63.64	58.56	64.65	53.65	78.48	75.26	78.68
Pair	D0→D3			D1→D3			D2→D3			D3→D3			D5→D3			all-but-D3→D3		
RNN	60.69	56.91	52.05	66.44	70.59	55.21	67.67	71.51	59.25	72.25	77.47	71.06	54.78	58.19	53.98	68.58	73.31	66.04
RNN+AE	65.36	67.70	59.07	68.53	72.84	60.15	67.14	69.98	56.21	72.00	79.68	72.46	61.33	60.20	52.50	73.17	77.09	72.05
Pair	D0→D4			D1→D4			D2→D4			D3→D4			D5→D4			all-but-D4→D4		
RNN	57.11	59.98	49.85	68.22	65.33	54.43	72.66	53.41	49.41	59.56	65.57	52.45	78.00	67.72	62.32	74.67	78.43	77.07
RNN+AE	77.56	65.77	55.25	81.56	67.29	52.10	79.11	51.39	51.36	81.11	66.97	58.86	77.78	69.21	55.98	79.33	77.31	78.44
Pair	D0→D5			D1→D5			D2→D5			D3→D5			D5→D5			all-but-D5→D5		
RNN	70.34	63.03	59.40	71.69	66.58	57.56	76.34	63.97	56.08	52.78	56.56	50.51	75.27	72.36	74.10	74.19	71.74	74.28
RNN+AE	75.54	70.13	59.65	74.46	73.24	50.82	77.42	68.83	58.38	73.48	70.56	60.54	77.15	76.83	74.91	77.42	73.04	74.45
Pair	D0→D6			D1→D6			D2→D6			D3→D6			D5→D6			all-but-D6→D6		
RNN	57.60	74.67	57.11	69.60	87.45	69.43	63.07	79.78	62.98	73.07	81.31	72.04	53.47	65.61	53.31	74.19	71.74	74.28
RNN+AE	65.73	80.62	65.15	71.33	92.19	71.18	57.60	82.52	57.38	69.07	86.93	68.71	56.00	77.45	55.71	78.00	90.30	78.14

Table 4: Cross-domain (e.g., D0→D1) and in-domain (e.g., D0→D0) results for CrisisMMD. The values that are statistically significant (according to a paired t-test with $p < .05$) for each pair are highlighted in bold font.

from one source disaster to the target disaster improves the results of the RNN classification network (especially, when the source and target disasters are similar, e.g., both are hurricanes), overall, the *all-but-one* models give the best results for a target, suggesting that a larger amount of training data results in better performance.

Based on our results, we can conclude that: 1) The domain adaptation with reconstruction approach (RNN+AE) has dramatically better performance than the domain adversarial with embeddings approach (GE+DA), in all metrics considered. This suggests that representation produced by the RNN-based seq2seq model is better than the representation obtained with the domain adversarial approach. 2) The domain adaptation with reconstruction has better performance than the cross-domain supervised RNN classification network in most cases, as it is specifically using unlabeled target data to bias the representation of the source towards the target. 3) Using all the available sources for training (*all-but-one* experiments) with the DRCN approach gives the best performance overall, suggesting that more data results in better models.

6 CONCLUSIONS

In this paper, we proposed a domain adaptation approach for disaster tweet classification. Experimental results show that the performance of proposed model is better than the performance of another adaptation model recently proposed for this task [1]. Given the promising results, our approach can potentially help eliminate the laborious process of labeling data in the beginning of a disaster. As a consequence, disaster response can be accelerated and resources can be better allocated. As part of future work, we plan to evaluate other implementations for the autoencoder, including Transformers, which have been shown to give good results in machine translation.

7 ACKNOWLEDGEMENTS

We thank the National Science Foundation and Amazon Web Services for support from grant IIS-1741345, which enabled the research and the computation in this study.

REFERENCES

- [1] 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)*. Association for Computational Linguistics, Melbourne, Australia, 1077–1087.
- [2] Firoj Alam, Ferda Ofli, and Muhammad Imran. 2018. CrisisMMD: Multimodal Twitter Datasets from Natural Disasters. In *Proceedings of the 12th International AAAI Conference on Web and Social Media (ICWSM)* (USA), 23–28.
- [3] Prakash Andugula and Surya S Durbha. 2015. Post-Disaster image analysis using domain adaptation. In *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. IEEE, 4848–4851.
- [4] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *JMLR* 17, 1 (2016), 2096–2030.
- [5] Muhammad Ghifary, W Bastiaan Kleijn, Mengjie Zhang, David Balduzzi, and Wen Li. 2016. Deep reconstruction-classification networks for unsupervised domain adaptation. In *European Conference on Computer Vision*. Springer, 597–613.
- [6] David Graf, Werner Retschitzegger, Wieland Schwinger, Birgit Pröll, and Elisabeth Kapsammer. 2018. Cross-domain informativeness classification for disaster situations. In *Proceedings of the 10th international conference on management of digital ecosystems*. 183–190.
- [7] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Comput.* 9, 8 (Nov. 1997), 1735–1780.
- [8] 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 (2015), 1–38.
- [9] Hongmin Li, Doina Caragea, Cornelia Caragea, and Nic Herndon. 2018. Disaster response aided by tweet classification with a domain adaptation approach. *Journal of Contingencies and Crisis Management* 26, 1 (2018), 16–27.
- [10] Hongmin Li, Oleksandra Sopova, Doina Caragea, and Cornelia Caragea. 2018. Domain Adaptation for Crisis Data Using Correlation Alignment and Self-Training. *International Journal of Information Systems for Crisis Response and Management (IJISCRAM)* 10, 4 (2018), 1–20.
- [11] Xukun Li, Doina Caragea, Cornelia Caragea, Muhammad Imran, and Ferda Ofli. 2019. Identifying Disaster Damage Images Using a Domain Adaptation Approach. In *16th International Conference on Information Systems for Crisis Response and Management (ISCRAM 2019)*.
- [12] Reza Mazloom, Hongmin Li, Doina Caragea, Cornelia Caragea, and Muhammad Imran. 2019. A Hybrid Domain Adaptation Approach for Identifying Crisis-Relevant Tweets. *International Journal of Information Systems for Crisis Response and Management (IJISCRAM)* 11, 2 (2019), 1–19.
- [13] Sinno Jialin Pan and Qiang Yang. 2009. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering* 22, 10 (2009), 1345–1359.
- [14] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in NIPS*. 3104–3112.
- [15] Carlos Villegas, Matthew Martinez, and Matthew Krause. 2018. Lessons from Harvey: Crisis Informatics for Urban Resilience. *Rice University Kinder Institute for Urban Research* (2018).