Disaster Response and Damage Assessment

Aditya Kakad
kakad@usc.edu

Pushkaraj Sarnobat
sarnobat@usc.edu

Shubham Nagarkar
slnagark@usc.edu

Abstract

During disasters, multimedia content on social media sites delivers vital information. Individuals utilize social media platforms like Twitter to report updates about injured or dead people, infrastructure damage, and missing or found people, among other types of information, during natural and man-made catastrophes. Reports of injured or deceased persons, infrastructure damage, and missing or found people are among the types of information shared. According to studies, this online information can be immensely beneficial for humanitarian groups in gaining situational awareness and planning relief activities provided it is handled quickly and properly. In this research, we propose employing state-of-the-art deep learning algorithms to create a joint representation from both text and image modalities of social media data. We use convolutional neural networks for image processing and BERT for text processing to define a multimodal deep learning system. We have employed an Early Fusion and Late fusion approach to concatenate the results obtained from both text processing and image processing pipelines to define results for our algorithm and to give us a substantial boost in results compared to baselines which were work done by early approaches. With best in class networks and efficient fusion techniques, we were successfully able to surpass the existing baselines in unimodal as well multimodal set up by a great margin.

Introduction

An increasing number of people use Social Media (SM) platforms like Twitter and Instagram to report information about critical emergencies or disaster events. Multimodal data shared on these platforms often contain useful information about the scale of the event, victims and infrastructure damage. This data can provide local authorities and humanitarian organizations with a big picture understanding of the emergency. Moreover, it can be used to effectively and timely plan relief responses.

The assessment of post-disaster damage is crucial for accounting for critical information such as the number of persons missing, injured, deceased, and so on, as well as damage to infrastructure and public property. This information is crucial in directing government funding and attention to the most severely affected communities in order to provide compensation, food, utilities, and emergency medical treatment. Assessment of damage provided by these algorithms can help organisations dispatch relief efforts to areas/people affected the most. These algorithms can greatly alleviate the relief effort which organisations can provide in these situations.

Our system aims to assist organisations by providing information about social media content posted by individuals in the affected area. This information ranges from alerting the authorities if a certain tweet is informative or not informative regarding the disaster, and what kind of humanitarian category a certain disaster lies in. The humanitarian categories are defined by their severity caused by the damage they induce. The humanitarian categories include affected individuals, infrastructure and utility damage, injured or dead people and others defined later in the paper. We also aim to achieve how severe the damage is from the images provided by users on Twitter by passing the images through a convolutional neural network and classifying the severity of the damage, which can range from severe, mild, and little/no damage caused.

Feature and decision-level fusion, also known as early and late fusion, is a common strategy for dealing with multimodality. In most cases, Multimodal approaches are combined by a distinct type of networking design which can be static, dynamic or N-Way classification at the hidden layers in deep learning topologies [9, 10, 11]. Despite substantial studies focusing primarily on social media text communications, there has been little effort in using visuals to increase humanitarian relief. The lack of ground-truth data is one factor impeding the advancement of this study path.

We have structured the remainder of the paper in the following ways - In the upcoming section, we provide a brief summary of the literature survey and related work conducted in this area. We then provide comprehensive details about the data used for our approach and the events of disaster where our studies focus on. We also provide information about the humanitarian tasks and the damage severity task performed in the following sections. We highlight the data preprocessing conducted by us on this data, and why this preprocessing was conducted. We then elaborate on our algorithmic effort as well as our fusion approaches.

Related Work

A number of studies have been conducted on social media data to help aid humanitarian efforts, and most of these studies have been conducted on social media textual data. With the recent boom in deep learning technologies being used, some approaches to analyse images related to disasters have been implemented as well. As noted by Bica et al. 2017 [21] in their study of social media photographs posted during two significant earthquakes in Nepal in April-May 2015, combining textual and visual content might yield extremely useful information. Their research focused on locating geo-tagged photographs and the damage they caused [13] looked at the relationship between tweets and photos, as well as how they might be used to distinguish visually relevant and irrelevant tweets. They created classifiers that combined text, image, and socially relevant contextual data (e.g., posting time, follower ratio, amount of comments, retweets) and reported an F1-score of 70.5% in a binary classification job, which is 5.7% higher than a text-only classification task.

Deep learning-based techniques such as Convolutional Neural Networks (CNN) [12] and Long-Short-Term-Memory Networks (LSTM) [17] have been frequently employed for the tweet categorization problem. State-of-the-art works for image categorization also employ deep neural network approaches such as Convolutional Neural Networks (CNN) with deep architectures. VGG [18], AlexNet [19], and GoogLeNet are the most common CNN designs [20]. The VGG is built on architecture with very small (33) convolution filters and a 16 to 19 layer depth.

Fusion strategies for datasets of many modalities have been used in a significant amount of multimedia research. The dynamics of the features formed are influenced by the type of fusion. The intra-modal interactions are poorly captured by early fusion techniques that are based on the simple concatenation. Late fusion approaches, on the other hand, place a premium on intra-model learning capacities while sacrificing cross-differentiability presented adversarial representational learning and graph fusion networks for multi-modal fusion to overcome these limitations.

Data

The **CrisisMMD Multimodal Twitter dataset** consists of several thousands of manually annotated tweets and images collected during seven major natural disasters including earthquakes, hurricanes, wildfires, and floods that happened in the year 2017 across different parts of the World. The provided datasets include three types of annotations:

1. **Informative vs Non-informative:**
   The purpose of this task is to determine whether a given tweet text or image, collected during a disaster event, is useful for humanitarian aid purposes. If the given text (image) is useful for humanitarian aid, it is considered as an "informative" tweet (image), otherwise as a "not-informative" tweet (image) [1].

2. **Humanitarian Categories:**
   The purpose of this task is to understand the type of information shared in a tweet text/image, which was collected during a disaster event. Given a tweet text/image, the task is to categorize it into one of the following categories;

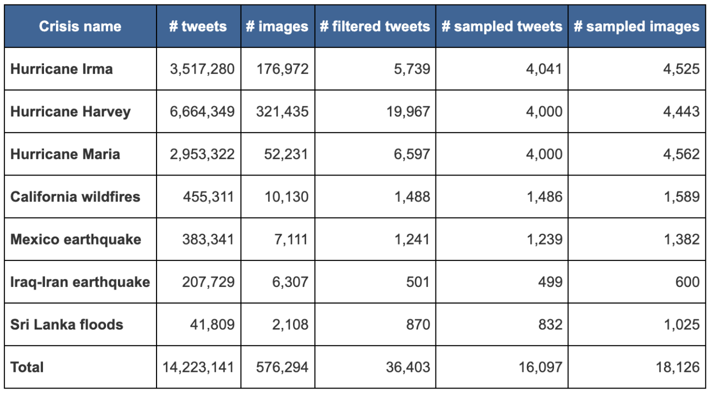
   - Affected individuals
   - Infrastructure and utility damage
   - Injured or dead people
   - Missing or found people
   - Rescue, volunteering or donation effort
   - Vehicle damage
   - Other relevant information

3. **Damage Severity Assessment**:
   The purpose of this task is to determine the severity of the damage caused to the infrastructure and utilities only. This task is associated only with the images and not the text. Each image belonging to the *infrastructure and utility damage* category is further classified into one of the following categories:

   - Severe Damage
   - Mild Damage
   - Little/No Damage

Data Exploration and Statistics: [dataset]

The seven catastrophes listed in the CrisisMMD dataset are shown in Table 1. The images and tweets in this dataset were collected from the social media website Twitter using hashtags associated with each occurrence, such as #HurricaneIrma, #HurricaneHarvey, and so on. After filtering extraneous noise from the scraped data and preserving just tweets with accompanying images, there are 18,126 sampled images and 16,097 sampled tweets. Because each tweet contains one or more images, the sampled images exceed the tweets.



Crisis name	# tweets	# images	# filtered tweets	# sampled tweets	# sampled images
Hurricane Irma	3,517,280	176,972	5,739	4,041	4,525
Hurricane Harvey	6,664,349	321,435	19,967	4,000	4,443
Hurricane Maria	2,953,322	52,231	6,597	4,000	4,562
California wildfires	455,311	10,130	1,488	1,486	1,589
Mexico earthquake	383,341	7,111	1,241	1,239	1,382
Iraq-Iran earthquake	207,729	6,307	501	499	600
Sri Lanka floods	41,809	2,108	870	832	1,025
Total	14,223,141	576,294	36,403	16,097	18,126

Table 1: Disaster types and sampled data points

Dataset Pre-Processing:

The sampled tweets and images described in the previous section cannot be directly used for our problem statement. We performed a series of data preprocessing steps to keep relevant data by removing noise and data imbalance. The different pre-processing techniques applied are as follows:

a. Label Matching:

Each tweet and the associated image is assigned to one of the eight classes of the Humanitarian categories. However, there are certain data samples with disagreeing labels between the images and tweets. In this step, we remove the tweets and the associated images that have different labels. Figure 1 shows the dataset size for train, validation and test split before and after *Label Matching*.

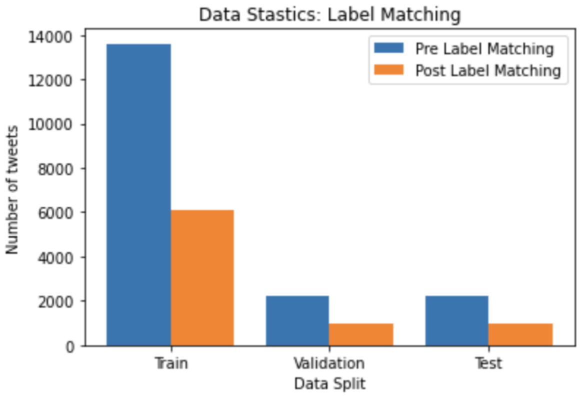
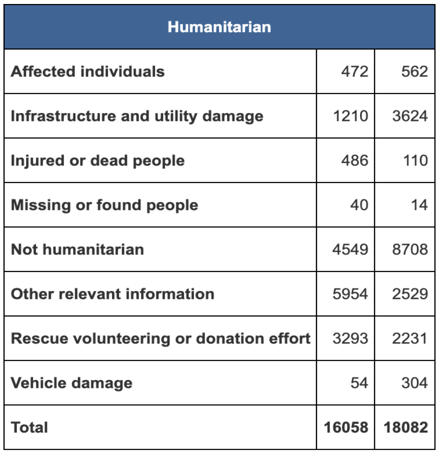


Figure 1: Data preprocessing - Label Matching.

b. Feature Engineering:

Table 2 denotes the data split between both tweets (left column) and images (right column) per class for the *Humanitarian Category* task. It is pretty evident from figure 2 that the number of data samples for each class/category is distributed unevenly. Any deep learning/ machine learning model trained on such an imbalanced and biased dataset will perform poorly irrespective of hyperparameter optimization.



Humanitarian		
Affected individuals	472	562
Infrastructure and utility damage	1210	3624
Injured or dead people	486	110
Missing or found people	40	14
Not humanitarian	4549	8708
Other relevant information	5954	2529
Rescue volunteering or donation effort	3293	2231
Vehicle damage	54	304
Total	**16058**	**18082**

Table 2: Humanitarian Category task data distribution.

In order to overcome the imbalance in this dataset, we performed a feature engineering step wherein we combined the categories - *affected_individuals, missing_or_found_people, and injured_or_dead_people* into one single category - *affected_individuals.* Similarly, we combined *infrastructure_and_utility_damage* and *vehicle_damage* into *infrastructure_and_utility_damage.*

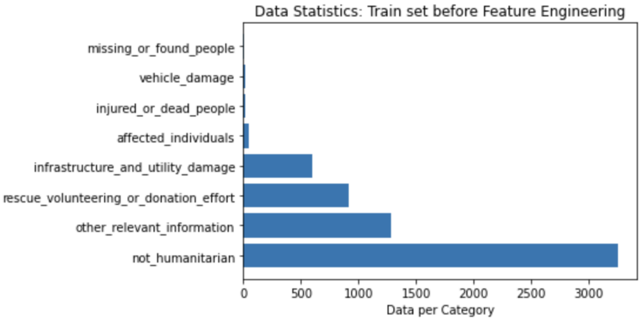


Figure 2: Data statistics before Feature Engineering

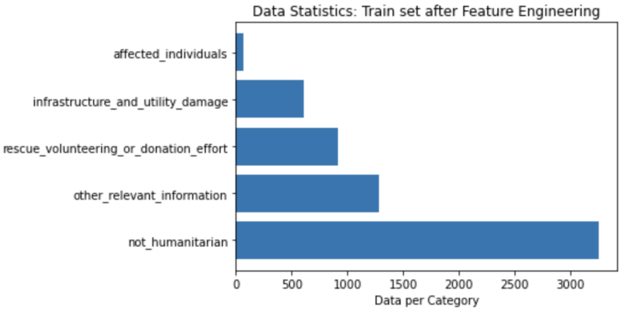
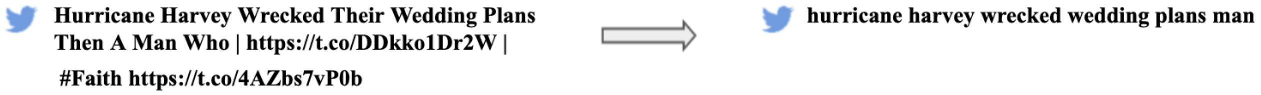


Figure 3: Data statistics after Feature Engineering

c. **Image preprocessing:** The scraped images come in a variety of sizes and formats. As a result, the first step in image preprocessing is to convert all images to 'png' format and resize them to 224x224 pixels. The rationale for selecting this size is that the majority of deep learning pre-trained models have an input size of 224x224. To avoid difficult calculations, the second step is to scale the pixel values from 0-255 to 0-1. Finally, these pixel values are normalized such that each pixel has the same data distribution.

d. **Text preprocessing:** The scraped tweets contain various hashtags and URLs along with some information textual data. In the text preprocessing step, we remove the hashtags, URLs, numbers, non-ASCII characters and stop-words like- *the, are, is, of,* etc. These characters do not add any significant meaning to the tweets and often mislead the deep learning models. Additionally, we have used contraction maps to replace shorthand words like “ya’ll" with “you all”. Since in any disaster situation there is a lot of commotion and panic amongst the people, at times people might have a lot of spelling mistakes in their tweets. These spelling mistakes were also found in the dataset where the words like firee (Fire), erthqke (Earthquake), damage (Damage) etc. were corrected using the Viterbi algorithm and Wikipedia word map. The words like Fire, Earthquake, Damage are keywords to identify and classify the tweet so it is very important to do this activity. Furthermore, we replaced all the punctuations with white spaces and converted the entire tweet into the lowercase format. Finally, since one tweet may have multiple images there are a significant amount of duplicate textual tweets present in the dataset, we remove all the duplicates and train the model only on the unique tweets.



Hurricane Harvey Wrecked Their Wedding Plans Then A Man Who | https://t.co/DDkko1Dr2W | #Faith https://t.co/4AZbs7vP0b

hurricane harvey wrecked wedding plans man

Figure 4: Text preprocessing on a random tweet.

Tasks and Approaches

In this study, we are focusing on the two most important tasks: Humanitarian Category and Damage Severity classification. Since we are dealing with a multimodal dataset, we have divided our approach into two pipelines viz. Text Pipeline is responsible for tweets classification and Image Pipeline is responsible for image classification.

Task 1: Humanitarian Category Classification

As mentioned earlier in the Data section, in this task we are trying to classify the type of information conveyed through the images and tweets. The output variables for this task are the 5 classes namely *affected_individuals, infrastructure_and_utility_damage, rescue_volunteering_or_donation_effort, other_relevant_information* and *not_humanitarian.* As the output classes are categorical in nature, this task is posed as a **Multi-class Classification** problem.

For this task, we perform three classification experiments where we train models using (i) only tweet text, (ii) only tweet image, and (iii) tweet text and image together. In the following subsections, we describe the deep learning approaches and architectures used for each modality as well as their training details.

Unimodal: Text

Deep learning has improved the performance of neural network architectures such as recurrent neural networks (RNN and LSTM) and convolutional neural networks (CNN) in solving a variety of Natural Language Processing (NLP) tasks such as text classification, language modelling, machine translation, and so on. One major issue is that RNNs can not be parallelized because they take one input at a time. In the case of a text sequence, an RNN or LSTM would take one token at a time as input. So, it will pass through the sequence token by token. Hence, training such a model on a big dataset will take a lot of time. As a result, we use transfer learning for our problem definition wherein we acquired larger pre-trained models and used these models as our good starting point of training.

To begin with, we experimented with our problem definition with a pre-trained BERT [14] model, opener's GPT-3 [15] and ELMo [16] model. At the end of the experiments, it was observed using grid search that the Unimodal Training is efficient when trained on pre pre-trained BERT model. Hence we selected the BERT model for the process of transfer learning. We used the Huggingface's transformers library to load the pre-trained uncased BERT model that has 110 million parameters. We then used the sklearn library to split the dataset into the train (70%), validation (15%) and test (15%).

Following is a code snippet and understanding of the tokenization of the text using the BERT model

```
# sample data
text = ["kakenews california wildfires destroy more than structures kakenews", "why california wildfires are worse the fall"]

# encode text
sent_id = tokenizer.batch_encode_plus(text, padding=True, return_token_type_ids=False)
```

Output:

```
{'input_ids': [[101, 10556, 7520, 7974, 2015, 2662, 3748, 26332, 6033, 2062, 2084, 5090, 10556, 7520, 7974, 2015, 102],
[101, 2339, 2662, 3748, 26332, 2024, 4788, 1996, 2991, 102, 0, 0, 0, 0, 0, 0, 0]],

'attention_mask': [[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1], [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0]]}
```

As you can see, the result is a two-item dictionary.

- The integer sequences of the input sentences are stored in 'input ids.' Special tokens are the integers 101 and 102. We append them to both sequences, with 0 being the padding token.
- 'Attention mask' is made up of 1s and 0s. It instructs the model to focus on the tokens with mask values of 1 and disregard the rest.

We used padding to make all of the messages the same length because the textual tweets in the dataset are of variable length. We looked at the distribution of text lengths in the train set to determine the correct padding length for our input text to the model.

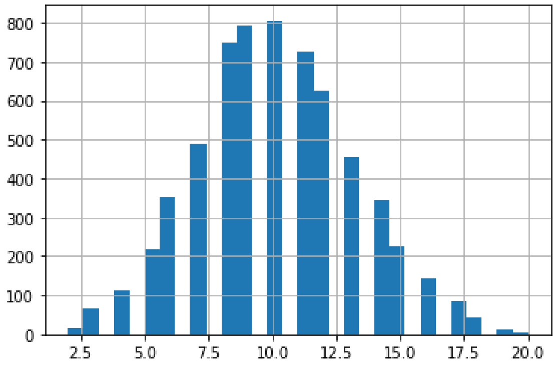


Figure 5: Overall distribution of length of tweets

The majority of the tweets had a length of 10 words or fewer, as can be seen. The greatest length is 20, while the minimum is 10. So, if we choose 20 as the padding length, all of the input sequences will be 20 characters long, and the majority of the tokens in those sequences will be padding tokens, which will not help the model learn anything meaningful and will also slow down the training. As a result, the padding length will be set to 10.

We constructed data loaders for both the train and validation sets after converting the messages in the train, validation, and test sets to integer sequences of length 10 tokens each. During the training phase, these data loaders sent batches of train and validation data to the model. Because batch size is a hyperparameter, it was discovered that a batch size of 32 produced the greatest results on the classification task. To fine-tune the BERT model, we added a classification head to the existing BERT model and trained it from the ground up while the BERT model remained frozen. Our classification head is a small DNN with a fully connected layer of 768 neurons, a fully connected layer of 512 neurons, and five softmax classification logits. The Representation Layer of the model is the intermediate fully connected layer with 512 neurons. As previously stated, our dataset has a class imbalance. We generated class weights for the labels in the train set first, and then passed these weights to the weighted loss function, which handled the class imbalance. AdamW was the optimizer we utilized. It's a better version of Adam's optimizer. Finally, we fine-tune our model with a batch size of 32, a learning rate of 10^{-3}, and a total of 30 epochs. During the training process, it was discovered that the loss decreases steadily, starting at 1.60 and ending at 0.84 by the 30th epoch for training samples and for the validation set the loss dipped to 0.984.

Unimodal: Image

For the imaging modality, we employ a **transfer learning** approach, which is an effective approach for visual recognition tasks [4]. The transfer learning strategy is based on using existing weights from a pre-trained model to give the model prior information on the classification problem. To train our model, we use the weights of a ResNet-50 model [5] that was previously trained on ImageNet. The idea behind employing a pre-trained CNN is to make use of the network's capacity to recognize lower-level characteristics such as edges, corners, lines, and so on. It saves a significant amount of time and computational effort against training the network from start. Furthermore, the dataset employed in this experiment is limited. As a result, the model will have fewer samples to train on in order to accurately detect low and high-level characteristics.

For this task, we carried out experiments with the ResNet-50 convolutional neural network. We replaced the last layer (i.e., softmax layer) of the network with a dense layer of 512 dimensions. This 512 dimensions dense layer acts as our robust feature representation layer that encodes all the information present in the image in a smaller space. On top of this layer, we further attach a classification head that consists of a simple multi-layered perceptron architecture consisting of a dense layer followed by batch-normalization, ReLU activation layer and a softmax layer instead of the original 1,000-way classification. The transfer learning approach allows us to transfer the features and the parameters of the network from the broad domain (i.e., large-scale image classification) to the specific one, in our case humanitarian classification task. Hence the output of this softmax layer has 5 neurons corresponding to the 5 classes of the humanitarian category.

The image model is trained using the Adam optimizer [6] with a batch size of 128 and an initial learning rate of 104, which is decreased by a factor of 0.1 when the loss on the development set stops falling by using PyTorch's ReduceLRonPlateau function. To avoid overfitting, we limited the number of epochs to 500 and used an early-stopping condition. Given the dataset's sparsity and imbalance, we use image augmentation techniques such as random horizontal flip and colour jitter to change brightness, contrast, saturation, and hue, in order to enhance model performance by decreasing overfitting.

Multimodal: Text + Image

Information from a single source is adequate but it is better to get additional information from multiple sources. There are multiple sources of data for a single problem, in our case images and tweets. These sources offer complementary information that not only helps to improve the performance of the model but also enables the model to learn better feature representations by utilizing the strengths of individual modalities. For instance, textual information is very sparse, whereas visual information from images for the same is more expressive. Combining these two gives us enriched information about the scene at hand. We attempt to employ this intuition by exploring early and late fusion techniques [7, 8] to achieve robust performance.

1. **Early Fusion:**

   Data-level fusion is another name for early fusion. The goal of this method is to merge the modalities at the data/feature stage. By running the images through the image pipeline, we turn them into embeddings (image representations) of 512 dimensions. Similarly, we use BERT to transform the tweets into 512-dimensional embeddings. As illustrated in Figure 5, we concatenate both of these embeddings to generate a 1024-dimension shared representation. The sizes of the embeddings are chosen to be equal in order to give equal weightage and value to the information gained from both texts and images. This shared representation is then followed by another multi-layered perceptron classification head with the output softmax layer with 5 neurons. Since the unimodal pipelines are already trained, in this approach we just aim to train the shared representations along with the MLP classification head using Adam optimizer, learning rate schedulers and early stopping technique.

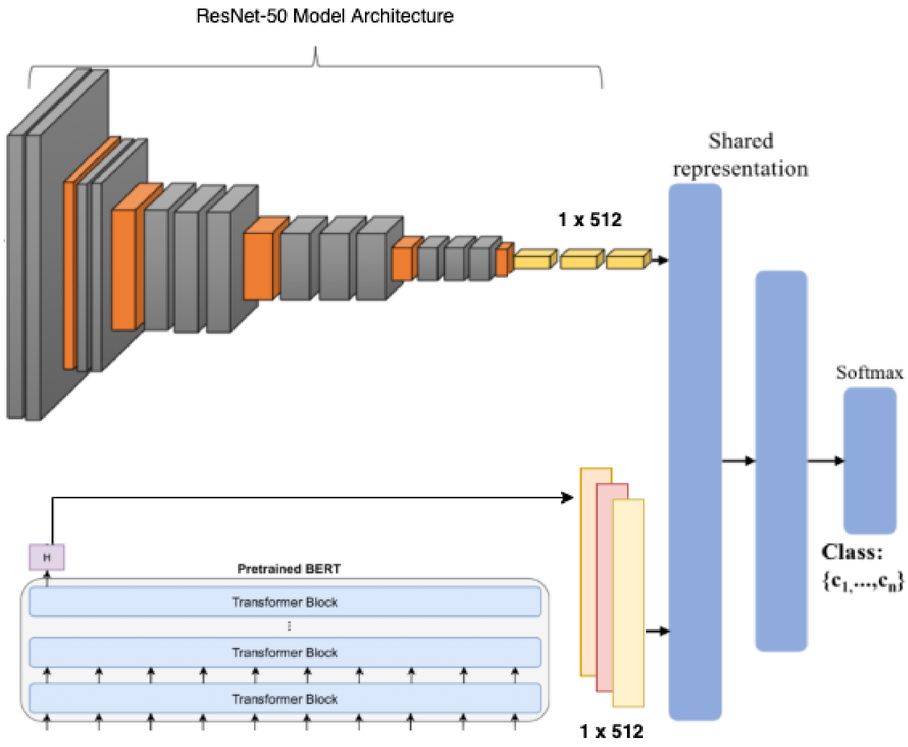


Figure 6: Multimodal architecture for Early Fusion technique on images and texts.

2. **Late Fusion:**
   Late Fusion is also known as Decision-level fusion as the fusion process takes place at the decision stage. The goal of this method is to fuse the decisions made by both the trained unimodal pipelines. This methodology is analogous to ensemble models. Once we have the class prediction based probabilities from both text and image representations, we further concatenate them and use different policy systems to evaluate which systems are able to best filter the required information. One of the important reasons to use late fusion is that it is not affected by the bias and correlation present in the data of both modalities. Figure 6 shows the basic outline of the late fusion approach.

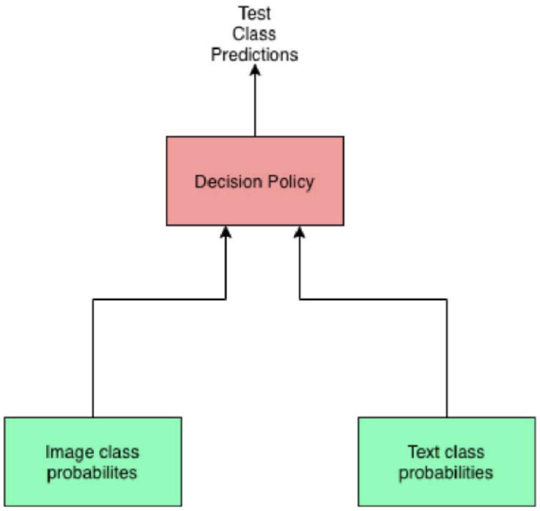


Figure 7: Multimodal architecture for Late Fusion technique on images and texts.

Task 2: Damage Severity Classification

For this task, we only select a subset of data that belongs to the *infrastructure_and_utility_damage* humanitarian category. These severity class labels are solely associated with images and not with text because images, as opposed to text, convey clear information about the damage inflicted by disasters. This task is purely dependent on anticipating the degree of the damage caused by catastrophic occurrences to infrastructure, vehicles, and utilities. This task comprises three degrees of damage severity viz. *Severe, Mild, and Little/No*. As a result, this is a **Multi-class Classification** problem as well.

For this task, we propose to conduct two different sets of experiments, i) training model from scratch and ii) using a trained model from task 1. We further intend to assess the results of both experiments to see if a model pre-trained on disaster data can forecast severity better than a purely randomly initialized model.

Results and Discussion

In this section, we present our results of the unimodal pipelines and compare these results with the baseline models [1, 3]. We also present our training and validation losses and accuracy curves. Since this is a multi-class classification problem, we use various metrics like accuracy, precision, recall and f1-score to evaluate the performances of our models.

Text: BERT

Because we employed a pre-trained model, we were able to train our classification model in just 30 epochs. Both the validation and training loss curves exhibited a continuously decreasing trend. However, after the 17th epoch, the model began to somewhat overfit due to data imbalance and sparsity. As a result, after the 17th epoch, we can see in Figure 7 that there is a disparity between the training loss and the validation loss. In terms of accuracy, the model's training, as well as validation accuracy, improves over time. Due to the data imbalance, accuracy is not a good statistic to use to evaluate the model's performance. Figure 8's confusion matrix clearly demonstrates this. Class A, which includes *affected_individials*, has been misclassified, with a 56 % accuracy rate. The *not_humanitarian, infrastructure and utility damage*, and *other relevant information* classes are appropriately classified by the model.

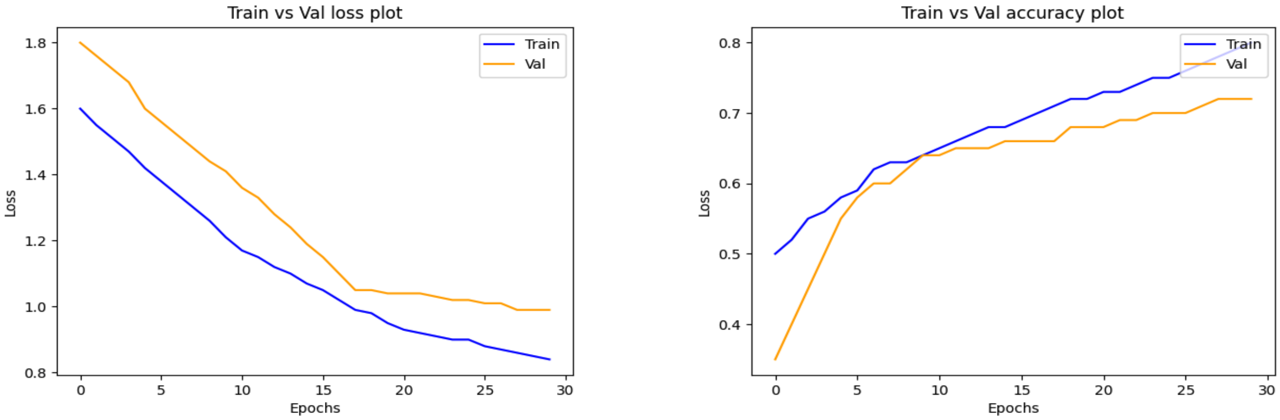


Figure 8: Training and Validation loss curves (left) and Training and Validation Accuracy curves (right) for BERT on the Humanitarian Category Classification task for Text

.

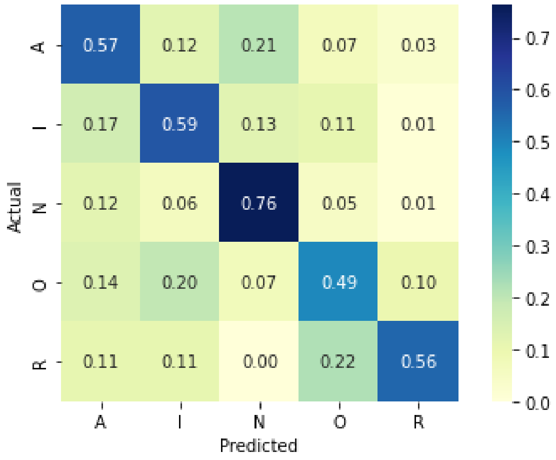


Figure 9: Confusion matrix of BERT on Humanitarian Category Classification task for text.

Image: ResNet-50

As we used the pre-trained networks in our pipeline, the model was trained very quickly within 10 epochs. The validation and the training loss both showed a gradually decreasing trend in their curves. However, due to data imbalance and sparsity, the model began to overfit. Hence we can see in figure 8 that there is a gap between the training loss and the validation loss. The training, as well as the validation accuracy of the model, is gradually increasing with the epochs. However, accuracy is not a good metric to evaluate the performance of the model due to the data imbalance. This is very evident in the confusion matrix of figure 9. Class A i.e. *affected_individials* have been misclassified and the accuracy of that particular class is 0%. The model does a great job by correctly classifying the *not_humanitarian, infrastructure_and_utility_damage* and *other_relevant_information* classes. The results shown are not our final results because they were acquired before class balancing strategies and a weighted loss function were implemented.

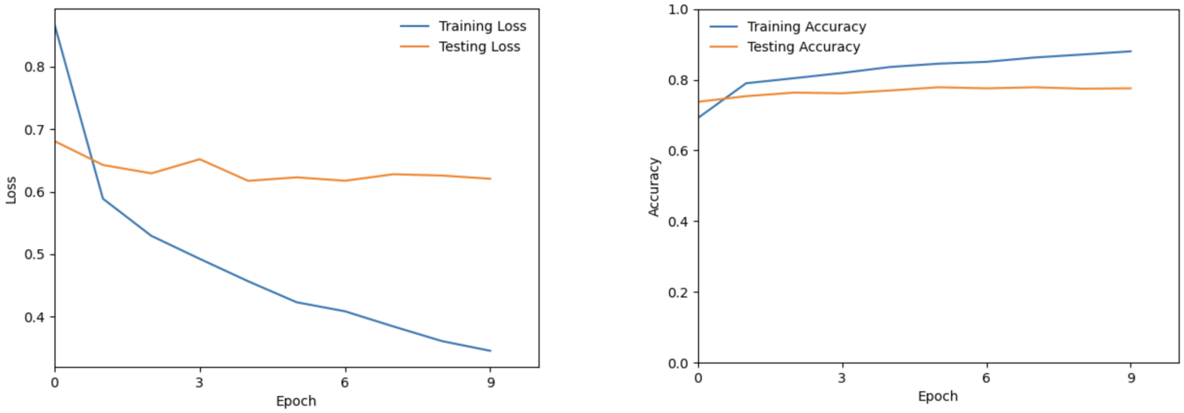


Figure 10: Training and Validation loss curves (left) and Training and Validation Accuracy curves (right) for ResNet-50 on the Humanitarian Category Classification task for images.

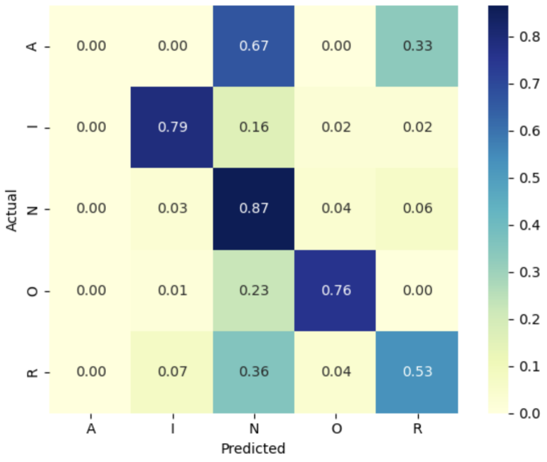


Figure 11: Confusion matrix of ResNet-50 on Humanitarian Category Classification task for images.

Table 3 depicts the comparison between the performance of models trained on text and image modalities for the humanitarian category classification task. It is clear that our models outperform the respective baseline models in both text and image modalities. Another important aspect to note here is that the image model outperforms the text models by a wide margin.

Modality	Model	Accuracy	Precision	Recall	F1-score
Text	Baseline	70.4	70.0	70.0	67.7
	BERT	72.93	71.0	71.23	69.9
Image	Baseline	76.8	76.4	76.8	76.3
	ResNet-50	78.0	77.4	78.0	77.3

Table 3: Model evaluations and baseline comparison for the Humanitarian Classification task.

Conclusion and Future Scope

Important informational signals gleaned from many data modalities on social media can be extremely beneficial to humanitarian organizations in disaster response. Despite the fact that photographs shared on social media contain useful information, previous research has mostly concentrated on text analysis, let alone merging the two modalities to improve performance. We suggested in this paper to learn a hybrid representation of social media data utilizing both text and image modalities. There are three sorts of annotations included in the datasets: informative vs. non-informative, humanitarian categories, and damage severity categories. We also discussed a variety of humanitarian use cases and tasks that could be accomplished with these datasets if more stable and effective methods were established. We have demonstrated that our methods outperform the baseline precision achieved by several other attempts at this problem. We are yet to combine early fusion and late fusion approaches into our methodology, but the results achieved by just preprocessing the data in the correct way gave us distinct benefits over the earlier described method.

First and foremost, CrisisMMD datasets can be applied to any multimodal task that involves computer vision and natural language processing. For instance, one might attempt to learn a combined embedding space of tweet text and images that can be employed both for text-to-image and image-to-text retrieval tasks. Another multimodal implementation of CrisisMMD is the image captioning task, which includes learning a mapping from visual content to its textual description.

References

1. Ferda Ofli, Firoj Alam, & Muhammad Imran. (2020). Analysis of Social Media Data using Multimodal Deep Learning for Disaster Response.
2. Firoj Alam, Ferda Ofli, & Muhammad Imran. (2018). CrisisMMD: Multimodal Twitter Datasets from Natural Disasters.
3. A. K. Gautam, L. Misra, A. Kumar, K. Misra, S. Aggarwal and R. R. Shah, "Multimodal Analysis of Disaster Tweets," 2019 IEEE Fifth International Conference on Multimedia Big Data (BigMM), 2019, pp. 94-103, doi: 10.1109/BigMM.2019.00-38.
4. Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. (2014). “How Transferable Are Features in Deep Neural Networks?” In: Advances in Neural Information Processing Systems, pp. 3320–3328.
5. Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. (2015). Deep Residual Learning for Image Recognition.
6. Zeiler, M. D. (2012). “ADADELTA: an adaptive learning rate method”. In: arXiv preprint arXiv:1212.5701.
7. Kuncheva, L. I. (2004). Combining pattern classifiers: methods and algorithms. John Wiley & Sons.
8. Alam, F. and Riccardi, G. (2014). “Fusion of acoustic, linguistic and psycholinguistic features for Speaker Personality Traits recognition”. In: 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 955–959.
9. Chowdhury, S. A., Stepanov, E. A., Danieli, M., and Riccardi, G. (2019). “Automatic classification of speech overlaps Feature representation and algorithms”. In: Computer Speech and Language 55, pp. 145–167
10. Nagrani, A., Albanie, S., and Zisserman, A. (2018). “Seeing Voices and Hearing Faces: Cross-Modal Biometric Matching”. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 8427–8436.
11. Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., and Ng, A. Y. (2011). “Multimodal deep learning”. In: Proceedings of the 28th international conference on machine learning (ICML-11), pp. 689–696.
12. [Nguyen et al. 2017] Nguyen, D. T.; Ofli, F.; Imran, M.; and Mitra, P. 2017. Damage assessment from social media imagery data during disasters. In International Conference on Advances in Social Networks Analysis and Mining (ASONAM), 1–8.
13. Chen, T., Lu, D., Kan, M.-Y., and Cui, P. (2013). “Understanding and classifying image tweets”. In: ACM International Conference on Multimedia, pp. 781–784.
14. Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova (2018). “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”
15. Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, Dario Amodei (2020). “Language Models are Few-Shot Learners”
16. Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, Luke Zettlemoyer (2018). “ Deep contextualized word representations”
17. Rosenthal, S., Farra, N., and Nakov, P. (2017). “SemEval-2017 task 4: Sentiment analysis in Twitter”. In: Proc. of the 11th SemEval, 2017), pp. 502–518.
18. Simonyan, K. and Zisserman, A. (2014). “Very deep convolutional networks for large-scale image recognition”. In: arXiv preprint arXiv:1409.1556.

19. Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). “Dropout: a simple way to prevent neural networks from overfitting.” In: Journal of MLR 15.1, pp. 1929–1958.
20. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). “Going deeper with convolutions”. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1–9.
21. Bica, M., Palen, L., and Bopp, C. (2017). “Visual Representations of Disaster.” In: Proc. of the CSCW, pp. 1262– 1276.