Multimodal Approaches for Disaster Tweets Classification

Gokul Nair gokuln@upenn.edu
Rohan Saraogi rsaraogi@upenn.edu
Shivani Prasad Bondapalli shivsspb@upenn.edu
Yash Agrawal yasha808@upenn.edu

Abstract

In this paper, we present our work on developing multimodal approaches for disaster tweet classification. We explore the use of text and image data to automatically classify tweets as informative or not informative for humanitarian aid purposes. Our research addresses the challenges of processing and analyzing social media data during disasters and leverages machine learning techniques to improve disaster response. We provide a literature review, experimental design, and results of our approaches, including extensions and error analysis. Our best performing model is an attention-based multimodal model that also integrates information about the specific disaster event name. It achieves a test accuracy of 0.787 and F-score of 0.773, noticeably beating our majority-class and unimodal baselines.

1 Introduction

In our digitally interconnected world, social media platforms play an important role in swiftly disseminating information, especially during critical events like natural disasters. Platforms such as Twitter serve as conduits for individuals to share a wealth of multimedia content, encompassing text and images. This content delivers real-time updates on various facets of unfolding situations, spanning casualties, infrastructure damage, the status of missing or found individuals, and more. The value of this online information for humanitarian organizations cannot be overstated. When processed promptly and efficiently, it significantly enhances their situational awareness, empowering them to plan and execute relief operations with greater effectiveness. This scenario presents a unique opportunity to develop a system that harnesses the combined potential of text and image data to automatically discern its relevance for humanitarian purposes.

To formally define the problem, we frame it as a classification task. Given a dataset of tweets and associated images, our objective is to develop a machine learning model that assigns each data instance to one of two classes: informative/not informative for humanitarian aid purposes. This binary classification problem necessitates the development of a robust and efficient system capable of processing both text and images to make informed decisions regarding the relevance of the multimedia content in the context of disaster response.

We have selected this task for our term project because we believe it holds both societal significance and academic interest. From a societal perspective, we believe that developing a system that can automatically identify valuable information within the deluge of social media data, can play a role in expediting aid delivery and saving lives. And from an academic standpoint, this task offers an exciting opportunity to explore the interplay between text and image modalities to help solve a classification task. We have limited prior experience with such tasks, and hence this can be a great learning opportunity for us.

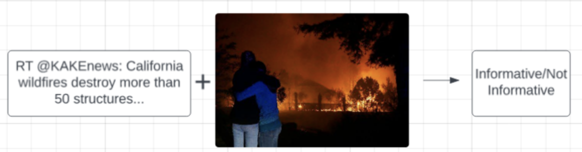

RT @KAKEnews: California wildfires destroy more than 50 structures...
+
Informative/Not Informative

Figure 1: Example tweet-image pair. Our goal is to build a system to classify the pair as "Informative"/"Not Informative" for humanitarian aid purposes.

2 Literature Review

The authors in (Alam et al., 2018) introduced the CrisisMMD: Multimodal Crisis Dataset and several associated tasks. We are working on Task 1, which is a binary classification task to determine whether a tweet-image pair is useful for humanitarian aid purposes (“Informative”) or not (“Not informative”). We provide further details regarding the dataset in the Experimental Design section.

There is some prior work addressing similar problems. The authors in (Ofli et al., 2020) employ a novel approach to the CrisisMMD dataset, combining CNNs for both text and image data from social media to enhance disaster response. They introduce a joint representation learning approach using two parallel deep learning architectures. For images, they use the VGG16 network architecture, extracting high-level features from images, and for text, they define a CNN with five hidden layers. These feature vectors from both modalities are then integrated into a shared representation, followed by a dense layer and a softmax prediction layer, a process known as early fusion. The team conducted experiments using three training settings: one using both text and image data, one using only text, and one using only images. The experimental results show that their multimodal approach significantly outperforms the models trained on a single modality (text or image alone). The best model in the multimodal setting achieved an F1-score of 0.842 for the informativeness classification task and 0.783 for the humanitarian classification task.

(Alam et al., 2017) presents Image4Act, an end-to-end pipeline for filtering and classifying Twitter images to support disaster response operations. It combines human computation and machine learning to handle noisy, high-volume social media data. Relevance filtering and deduplication are done using deep neural networks and perceptual hashing, achieving AUC/accuracy scores of 0.98. Additional custom classifiers for damage assessment are created via crowdsourcing. Evaluations show the damage classifier reaches an AUC of 0.72. Real-world deployment during a cyclone filtered images with 0.67-0.92 precision.

(Srinivasulu Kothuru, 2023) introduces a labelled data-efficient strategy to identify COVID-19 informative tweets with a limited number of labelled instances, achieving an F1-score of 0.912. The approach applies data augmentation to expand the training set, enhancing the performance of pre-trained language models with just 14.3% of the full dataset size. The authors address the challenges of attaining comparable performance with fewer data, the effectiveness of various augmentation methods, and the reasons for improved model efficiency through augmentation. Leveraging transformer-based models like CT-BERT, RoBERTa, and BERTweet, along with augmentation techniques such as AEDA, EDA, BT, and T5-based paraphrasing, the study shows that CT-BERT fine-tuned with on tweets nearly matches the performance achieved with 7000 tweets. This finding underscores the potential of their data-efficient method for situations where large-scale labelling is impractical.

3 Experimental Design

In this section we describe our data, evaluation metrics, and the performance of our simple baseline.

3.1 Data

The authors in (Alam et al., 2018) introduced the CrisisMMD: Multimodal Crisis Dataset and several associated tasks. We used version 2.0 of the dataset and are working on Task 1, which is a binary classification task to determine whether a tweet-image pair is useful for humanitarian aid purposes (“Informative”) or not (“Not Informative”). The 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 authors consider a tweet/image as “Informative” if it reports/shows one or more of the following: cautions, advice, and warnings, injured, dead, or affected people, rescue, volunteering, or donation request or effort, damaged houses, damaged roads, damaged buildings; flooded houses, flooded streets; blocked roads, blocked bridges, blocked pathways; any built structure affected by earthquake, fire, heavy rain, strong winds, gust, etc., disaster area maps. Images showing banners, logos, and cartoons are not considered as “Informative”.

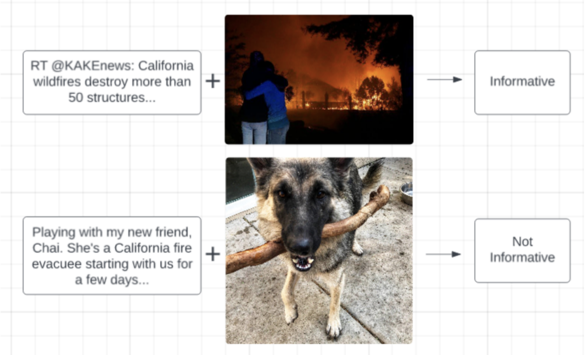
Figure 2 shows sample input-output pairs. And Table 1 shows the no. of samples and % of informative samples in the train/dev/test splits provided by the authors. We note that there is some skew in the data in favour of informative samples.

Table 1: Train/Dev/Test Split Sizes

	Train	Dev	Test
# Samples	13,608	2,237	2,237
% "Informative"	61.295	62.897	61.377

3.2 Evaluation Metric

We used evaluation metrics similar to what was used in (Ofli et al., 2020) (accuracy, precision, re-



RT @KAKEnews: California wildfires destroy more than 50 structures...

Playing with my new friend, Chai. She's a California fire evacuee starting with us for a few days...

Figure 2: Example "Informative"/"Not Informative" tweet-image pairs.

call, and F-scores). We also report the area-under-curve (AUC) scores.

3.3 Simple Baseline

For our simple baseline we used a majority-class classifier based on the majority label in the train data. The majority label is "Informative", and Table 2 shows the model performance. As can be seen, the model achieves a test accuracy and F-score of 0.614 and 0.380 respectively. The model performs poorly as the data distribution is the sole estimator of how the label for any future example is determined, without considering any other factors that would influence the prediction.

Table 2: Majority-Class Classifier Performance

Metric	**Train**	**Dev**	**Test**
Accuracy	0.613	0.629	0.614
Precision	0.306	0.314	0.307
Recall	0.500	0.500	0.500
F1	0.380	0.386	0.380
AUC	0.500	0.500	0.500

4 Experimental Results

In this section we describe the implementation and performance of our strong baseline and extensions. We also provide an error analysis of our best performing model.

4.1 Strong Baseline

We experimented with text-only and image-only baselines.

4.1.1 Text-Only Baseline

For the text-only baseline, we generated text embeddings using the `all-MiniLM-L6-v2` sentence-transformer (https://www.sbert.net/) and trained a logistic regression model with default parameters on it.

4.1.2 Image-Only Baseline

For the image-only baseline we fine-tuned the classification head of a pre-trained VGG16 model with ImageNet weights. In particular, we added a fully-connected layer (with 2-dimensional outputs) to the 1000-dimensional outputs of VGG16, with ReLU activation in-between. We trained the model using cross-entropy loss and an Adam optimizer with learning rate 1e-4 for 20 epochs. This configuration was similar to the unimodal image-only baseline adopted by the authors in (Ofli et al., 2020).

4.1.3 Results

The results for the strong baselines are shown in Table 3. We observe that both baselines show better test performance across all metrics compared to the majority-class baseline. Also, the text-only baseline performs noticeably better (with test accuracy and F-scores of 0.743 and 0.713 respectively) compared to the image-only baseline (with test accuracy and F-scores of 0.662 and 0.582 respectively). The noticeable jump in performance of the text-only baseline compared to the majority-class baseline suggests the value of sentence-transformer embeddings for our task.

The performance of the image-only baseline is noticeably worse compared to that of the authors in (Ofli et al., 2020). There could be several potential reasons for this, that make our results not directly comparable: (1) As mentioned previously, we are working with version 2.0 of the data. Looking at the data statistics provided by the authors, it seems like the authors are using a different version of the data. (2) The authors trained their model for 1000 epochs with an initial learning rate of 1e-6 and an early stopping criterion. Due to resource constraints, we were only able to train our model for 20 epochs and used a higher learning rate of 1e-4 to compensate for the fewer epochs.

4.2 Extension 1

For the first extension we compared the performance of two different embedding generation strategies. For both sets of embeddings we trained an elastic net model with different choices of `l1_ratio` $\in [0, 0.25, 0.5, 0.75, 1]$ to predict the output label and looked for the model that maximized the dev accuracy/F-score.

Table 3: Strong Baseline Performance

Model	Metric	Train	Dev	Test
Text	Accuracy	0.754	0.736	0.743
	Precision	0.748	0.721	0.738
	Recall	0.720	0.694	0.706
	F1	0.727	0.701	0.713
	AUC	0.806	0.781	0.790
Image	Accuracy	0.682	0.673	0.662
	Precision	0.691	0.659	0.652
	Recall	0.612	0.590	0.593
	F1	0.603	0.579	0.582
	AUC	0.729	0.681	0.702

4.2.1 CLIP-Concat

We used OpenAI's `ViT-B/32` CLIP (Contrastive Language-Image Pre-Training) model (https://github.com/openai/CLIP), a neural network trained on a variety of (image, text) pairs. In particular, we used the model to generate text and image embeddings for each tweet-image pair, and concatenated the generated embeddings. The system diagram is shown in Figure 3.

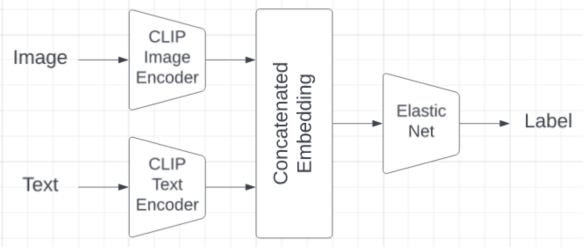


Figure 3: CLIP-Concat model

4.2.2 Text + Captions + S-BERT

We used Salesforce's BLIP (Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation) framework (https://github.com/salesforce/BLIP) to generate image captions using nucleus sampling. Given a tweet-image pair, and generated image caption `caption`, we augmented the tweet text as `Tweet: tweet_text\n Image Caption: caption`. Finally, we generated sentence-transformer (https://www.sbert.net/) embeddings for the augmented text using the `all-MiniLM-L6-v2` model. The system diagram is shown in Figure 4.

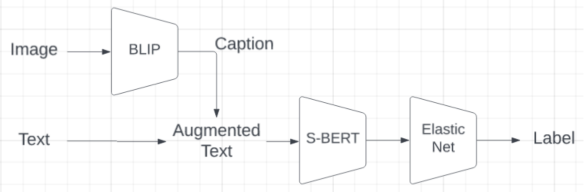


Figure 4: Text + Captions + S-BERT model

4.2.3 Results

Our experiments with different `l1_ratio` values had a marginal ($< 1\%$) effect on dev-accuracy and F-score for both embedding generation strategies, with `l1_ratio` = 0 showing the best results. The results for `l1_ratio` = 0 are shown in Table 4.

We observe that **CLIP-Concat** has consistently better performance than **Text + Captions + S-BERT**, and achieves a test accuracy and F-score of 0.785 and 0.77 respectively. This seems reasonable as, among other reasons, we would assume that there is more information in an image than just a caption for it, and CLIP would produce text and image embeddings that align well with each other. We also observe that both models have significantly better performance than our simple baseline (which had test accuracy and F-score of 0.614 and 0.380 respectively) and image-only strong baseline (with test accuracy and F-scores of 0.662 and 0.582 respectively), and also outperform our text-only strong baseline (with test accuracy and F-scores of 0.743 and 0.713 respectively).

Table 4: Extension 1 Performance

Model	Metric	Train	Dev	Test
CLIP-Concat	Accuracy	0.829	0.806	0.785
	Precision	0.822	0.793	0.774
	Recall	0.814	0.788	0.767
	F1	0.817	0.790	0.770
	AUC	0.909	0.869	0.860
Text + Captions + S-BERT	Accuracy	0.783	0.759	0.766
	Precision	0.778	0.746	0.758
	Recall	0.756	0.725	0.738
	F1	0.763	0.732	0.744
	AUC	0.850	0.829	0.837

Figure 5 shows some examples of the captions generated by the BLIP framework. Based on manual inspection of a sample of the captions generated, we were reasonably satisfied with their quality. This seems to bear out in the results, since **Text**

+ Captions + S-BERT has a 2-3% improvement in test accuracy and F-scores over the text-only baseline, which highlights the value of the image captions in the learning process.

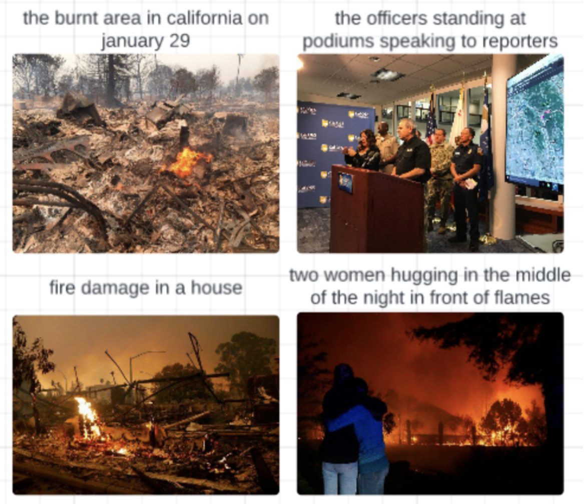

the burnt area in california on january 29
the officers standing at podiums speaking to reporters
fire damage in a house
two women hugging in the middle of the night in front of flames

Figure 5: Example image captions (above) and images (below) generated by the BLIP framework.

4.3 Extension 2

Given that **CLIP-Concat** had the best performance in Extension 1, we wanted to perform a further experiment with it using an attention-like framework. In addition, our dataset has a column with the event associated with a given tweet-image pair (california_wildfires, srilanka_floods, mexico_earthquake, iraq_iran_earthquake, hurricane_harvey, hurricane_maria, hurricane_irma). We wanted to incorporate this information in the training process. The resulting model diagram is shown in Figure 6.

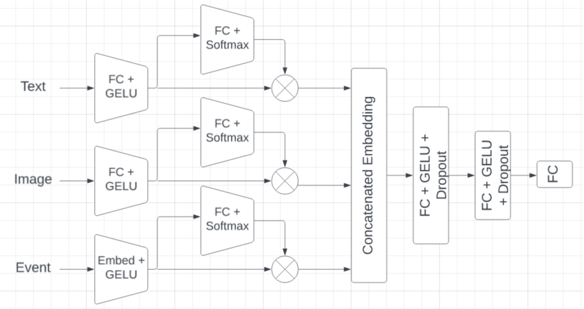


Figure 6: Attention Model

The model has three inputs: 512-dimensional CLIP text embeddings, 512-dimensional CLIP image embeddings, and 1-dimensional event ID (integer between 0-6 inclusive). The text and image embeddings are passed through fully-connected layers (with 256-dimensional outputs) and GELU activation. And the event ID is passed through an embedding layer (32-dimensional outputs) and GELU activation. The text, image, and event outputs are then each passed through separate fully-connected layers with softmax activation to generate attention weights for each dimension. The attention weights are then multiplied with the text, image, and event outputs. Finally, the results are concatenated into a 256 + 256 + 32 = 544-dimensional vector and passed through 2 fully-connected layers with GELU activation and dropout (p=0.5), and an output fully-connected layer (with 2-dimensional outputs).

We train the model with cross-entropy loss, an Adam optimizer with an initial learning rate of 1e-5, and with a `ReduceLROnPlateau` scheduler with a patience of 25. The train/dev loss and accuracy curves are shown in Figure 7.

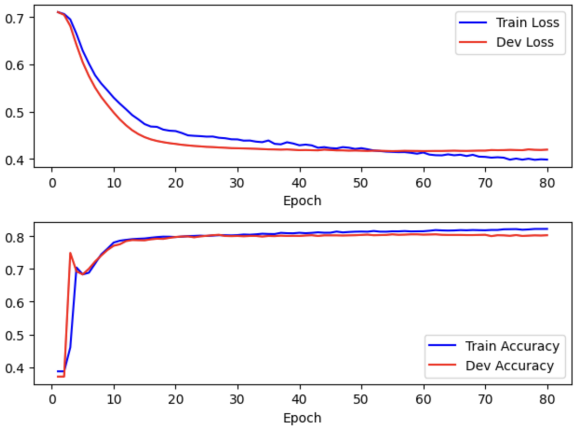


Figure 7: Attention Model Train/Dev Loss/Accuracy Curves

The model performance results are shown in Table 5. We observe that this framework did not noticeably improve performance as the test performance metrics are only marginally better than that of the **CLIP-Concat** model. We think there could be multiple potential reasons for this: (1) We are working with a limited dataset size (∼10000-20000 samples), which prevents us from experimenting with large model sizes without overfitting. In particular, we used pre-trained CLIP embeddings (without fine-tuning) and added a small attention model with regularization approaches like dropout and a small learning rate. Despite this, comparing the train and test performance metrics shows that there is some overfitting. A potential solution to this would be multi-modal data augmentation strategies which could be a future direction for this work (2) We attempted to use attention with the embedding dimensions. An alternative approach would be to use conventional sequence based approaches to attention such as on smaller sections of an image or words in a tweet. Experimenting with this approach could also be a future direction for this work.

Table 5: Extension 2 Performance

Metric	Train	Dev	Test
Accuracy	0.824	0.803	0.787
Precision	0.816	0.790	0.776
Recall	0.810	0.787	0.771
F1	0.813	0.788	0.773
AUC	0.903	0.881	0.875

4.4 Error Analysis

Our best performing model was our attention model from Extension 2. The test confusion matrix for this model is shown in Figure 8. We observe that there is a similar no. of false-positives and false-negatives. This is line with the balanced precision, recall and F-score values in Table 5.

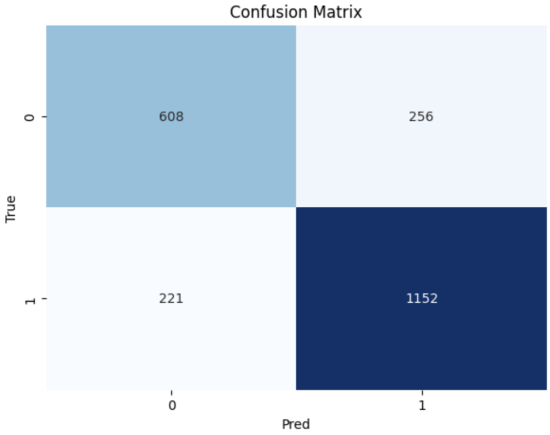


Figure 8: Attention Model Test Confusion Matrix

If we split the test samples based on the event, and look at the proportion of incorrect predictions for each event, we observe that the hurricane events (and `hurricane_irma` in particular) have the highest proportion of incorrect results. This is shown in Figure 9.

For a given text-image pair, our dataset contains informativeness labels for the individual text, individual image, and for the text-image pair. If the labels for the text and image are inconsistent (i.e. one is informative and the other is not), the label for the text-image pair is determined randomly. This is both a key limitation and a challenge with the way the authors have structured their data. To investigate Figure 9 further, Table 6 shows the proportion of inconsistent labels for each event. In particular, for a given event, we are looking at the ratio between the no. of test samples for which the text and image labels don't match and the total no. of test samples for the event.

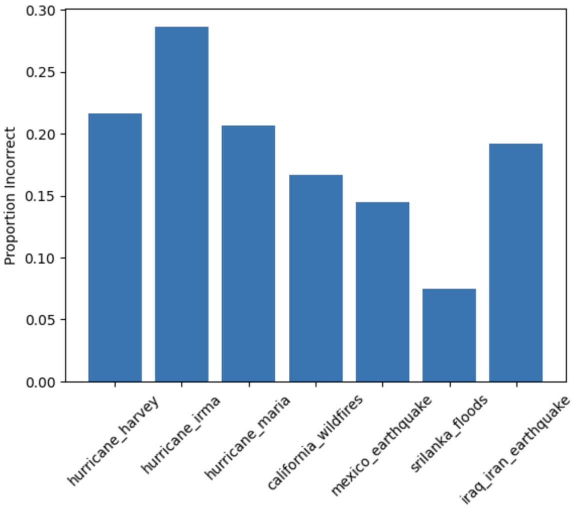


Figure 9: Event-wise proportion of incorrect test predictions for the attention model

Table 6: Proportion of test samples for which the text and image labels don't match for each event

Event	Proportion Inconsistent
hurricane_harvey	0.308
hurricane_irma	0.409
hurricane_maria	0.338
california_wildfires	0.254
mexico_earthquake	0.201
srilanka_floods	0.121
iraq_iran_earthquake	0.205

We observe that the hurricane events (and `hurricane_irma` in particular) are precisely the events with the largest proportion of inconsistencies. Also, in Table 7 we show the no. of incorrect test predictions for different combinations of text and image labels.

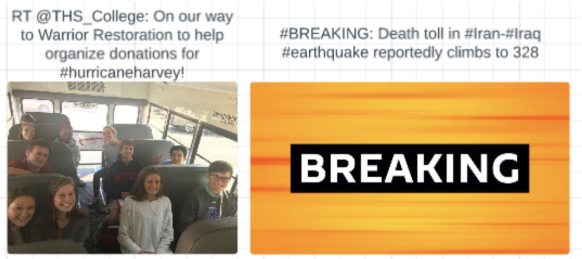
We observe that ∼75% of the incorrect test predictions are for cases with inconsistent text and image labels. Hence the text-image label inconsistency seems to be a major factor for our model's

Table 7: No. of incorrect test predictions for different combinations of text and image labels for the attention model

Text Label	Image Label	# Incorrect
Not Informative	Not Informative	67
Informative	Not Informative	299
Not Informative	Informative	62
Informative	Informative	49

incorrect predictions. This is natural, as inconsistent labels for the texts and images can be a clear source of confusion for the model, especially when the pair's label has been set randomly.

We show examples of cases with informative text and not-informative images for which the attention model predicted incorrectly in Figure 10.



RT @THS_College: On our way to Warrior Restoration to help organize donations for #hurricaneharvey!

#BREAKING: Death toll in #Iran-#Iraq #earthquake reportedly climbs to 328

Figure 10: Examples of cases with informative text and not-informative images for which the attention model predicted incorrectly

Finally, in Table 8 we compare the unimodal baselines with the attention model on the test data. In particular, we show the no. of cases for which a unimodal baseline is incorrect and the attention model is correct and vice versa. For these cases, we also show the % of cases for which the true label is informative.

Table 8: Comparison on test data of cases where unimodal baselines are incorrect and the attention model is correct and vice versa. The last column is the % of these cases for which the true label is informative.

Image-Only	Attention	Count	% Inform.
Incorrect	Correct	476	15.966
Correct	Incorrect	197	78.680
Text-Only	**Attention**	**Count**	**% Inform.**
Incorrect	Correct	294	32.993
Correct	Incorrect	197	71.066

Looking at Table 8, we observe that there are substantially more cases for which the image-only baseline was incorrect and the attention model was correct compared to the cases where the text-only baseline was incorrect and the attention model was correct. This makes sense as the text-only baseline performed noticeably better than the image-only baseline, and hence is more competitive with our attention model. In addition, it seems like the attention model is better than the unimodal baselines at identifying samples that are not informative. On manual inspection, it was not immediately obvious why this might be the case. Also, given the limited sample sizes, it would be difficult to draw any conclusions from this observation without substantial further exploration.

5 Conclusions

In our final project, we explored different approaches for disaster tweet classification, aiming to classify tweets as informative or not for humanitarian aid. We had the opportunity to experiment and compare the performances of different unimodal and multimodal models for our problem. Our best model, an attention-based multimodal model integrating specific disaster event names, achieved notable success with a test accuracy of 0.787 and an F-score of 0.773. This performance surpassed our majority-class and unimodal baselines. In spite of limited compute power and a limited dataset which are required for state-of-the-art models, our results were promising, demonstrating the potential of multimodal models in processing complex data in disaster scenarios. The project provided valuable insights into the challenges of analyzing social media for humanitarian purposes and the effectiveness of multimodal machine learning techniques.

A key pain point in our project was the limited size of our dataset (10000 $\sim$ 20000) samples. This limited our ability to experiment with larger models without the risk of overfitting. As such, a natural next step would be to experiment with data augmentation methods. Other than this, the attention approach that we used was to use attention for the individual embedding dimensions. Alternative approaches would be to look at attention on individual tweet tokens or subimages in the images. These are some possible directions for future work.

6 Acknowledgements

Special thank you to our project TA Yufei Wang for her valuable guidance throughout the duration of the project. We would also like to thank Prof. Mark Yatskar and all other course TAs for giving us a meaningful learning experience while taking this course.

References

Firoj Alam, Muhammad Imran, and Ferda Ofli. 2017. Image4act: Online social media image processing for disaster response. In *2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 601–604. IEEE/ACM.

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)*.

Ferda Ofli, Firoj Alam, and Muhammad Imran. 2020. Analysis of social media data using multimodal deep learning for disaster response. In *17th International Conference on Information Systems for Crisis Response and Management*. ISCRAM, ISCRAM.

A. Santhanavijayan Srinivasulu Kothuru. 2023. Identifying covid-19 english informative tweets using limited labelled data. In *Social Network Analysis and Mining*.