Summer Internship Report

On

Multimodal-Tweets-Classification using CrisisMMD Dataset

submitted by

Ravindra Kumar (17NA10023)

under the guidance of

Prof. Saptarshi Ghosh

Computer Science and Engineering Department,

Indian Institute of Technology,

Kharagpur-721302

ACKNOWLEDGEMENT

I would like to show my gratitude to Prof. Saptarshi Ghosh for giving me this internship and his constant support along with valuable guidance throughout the completion of the intern. I also would like to thank the department of “Computer Science and Engineering” for giving me this opportunity.

1. Introduction

In recent years, the world has been hit with a series of big natural disasters, including earthquakes, hurricanes, wildfires and floods across the different parts of it. During these difficult times social media often plays a key role to spread the information on damages occur in these disasters but a significant amount of it, is redundant, so it is very important to extract the valuable information from it such as reports of injured or dead people, critical infrastructure and utility damage, urgent needs of affected people, and missing or found people to facilitate required help to humanitarian organizations in preparedness, response, mitigation and recovery efforts. The information we get from social media in the modern digital world is multimodal in nature, textual information on the web rarely occurs alone, and is often accompanied by images, sounds, videos, or other modalities. Some past research works have demonstrated that text along with images shared on social media during a disaster event can help a deep learning model to perform better in classification tasks instead of using only text. So, in this work, I utilize both text and image modalities information available, for building a deep learning multimodal architecture using transfer learning. This architecture combines two features representation, extracted from the penultimate fully connected layers of two pre-trained unimodal text and image networks from raw text and images independently, using Feature fusion or EmbraceNet fusion method then pass this feature representation to terminal network. Many combinations of pre-trained text and image models have been used to construct the architecture using the above techniques. Unimodal pre-trained image models such as resnet50, resnet152, densenet161 and vgg19_bn for image data and unimodal pre-trained text models like AWD_LSTM, BERT, RoBERTa, XLNet, XLM, and DistilBERT have been used in this work.

2. Dataset

This CrisisMMD version 2.0 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. There are three classification tasks associated with the different classes. In this analysis, I focus only on two classification tasks given below using agreed labels for both image and text modalities.

The agreed labeled dataset is provided with three splits: train, development, and test separately for both the task, class-wise distribution is given below.

Task1. Classification 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).

Table1: Train, dev and test set class wise distribution for Informativeness Task.

	# of samples in respective sets			
Class	**Train**	**Dev**	**Test**	**Total**
Informative	6345	1056	1030	8431
Not Informative	3256	517	504	4277
Total	9601	1573	1534	12708

Task2. Classification into various 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 categories listed in Table below.

Tabel2: Train, dev and test set class wise distribution for Humanitarian Task.

	# of samples in respective sets			
Class	**Train**	**Dev**	**Test**	**Total**
Not humanitarian	3252	521	504	4277
Other relevant information	1279	239	235	1573
Rescue volunteering or another donation effort	912	149	126	1183
Infrastructure and utility damage	612	80	81	773
Affected individuals	71	9	9	89
Total	6126	998	955	8079

For further information on the dataset you can refer to this paper and visit the CrisisMMD data website. A sample of raw data with agreed labels for both modalities given below in the data frame

text	target
xxbos ya ll be sure to get in on this week xxunk ve got to know what you think you know in case need to know	not_humanitarian
xxbos update on my mom in dr and maria same path as irma eye bit to her north should be strong winds some rain but	not_humanitarian
xxbos the best part was that it let me spend time with my family early and they got to meet my new kitten sam xxunk	not_humanitarian
xxbos all eyes on maria as it heads over simliar islands that got hit with irma sending fto them as we watch it closely	other_relevant_information
xxbos all eyes on maria as it heads over simliar islands that got hit with irma sending fto them as we watch it closely	other_relevant_information

Figure1. Agreed label raw data sample

Data Pre-processing

The tweet text column of the agreed labeled dataset contains many redundant non-ASCII characters, numbers, URLs, and hashtag signs which have been removed before tokenization as it is found by experiment that these links, emoji and other characters are not helping the model to perform better.

Every time when a batch is fed to the model, instead of feeding same images, some small random transformations are applied to the images, such as a bit of rotation, zoom, resize, translation, flipping ,etc that don't change what's inside the image (to the human eye) but do change its pixel values, for this default values of hyperparameters in Fastai work pretty well.

Here is the text and image batch sample look like after initial pre-processing.

	event_name	tweet_id	image_id	tweet_text	image	label
5348	hurricane_irma	910239253715607552	910239253715607552_0	#Medshare sent supplies to Irma/Harvey victims...	data_image/hurricane_irma /19_9_2017/9102392537...	rescue_volunteering_or_donation_effort
1829	hurricane_maria	914663754117386240	914663754117386240_0	@JLo Just 1 pic of the damage done to my optom...	data_image/hurricane_maria /2_10_2017/914663754...	infrastructure_and_utility_damage
6790	california_wildfires	918273289197416448	918273289197416448_0	How You Can Help With the #California Fire Rel...	data_image/california_wildfires /12_10_2017/918...	not_humanitarian
5653	hurricane_irma	910146307884494848	910146307884494848_0	RT @paul_lander: Breaking Hurricane Irma news:...	data_image/hurricane_irma /19_9_2017/9101463078...	not_humanitarian
3141	hurricane_maria	912130173541273600	912130173541273600_0	In addition to #HurricaneMaria we also have #L...	data_image/hurricane_maria /25_9_2017/912130173...	other_relevant_information

Figure2. Pre-processed Text Batch Sample

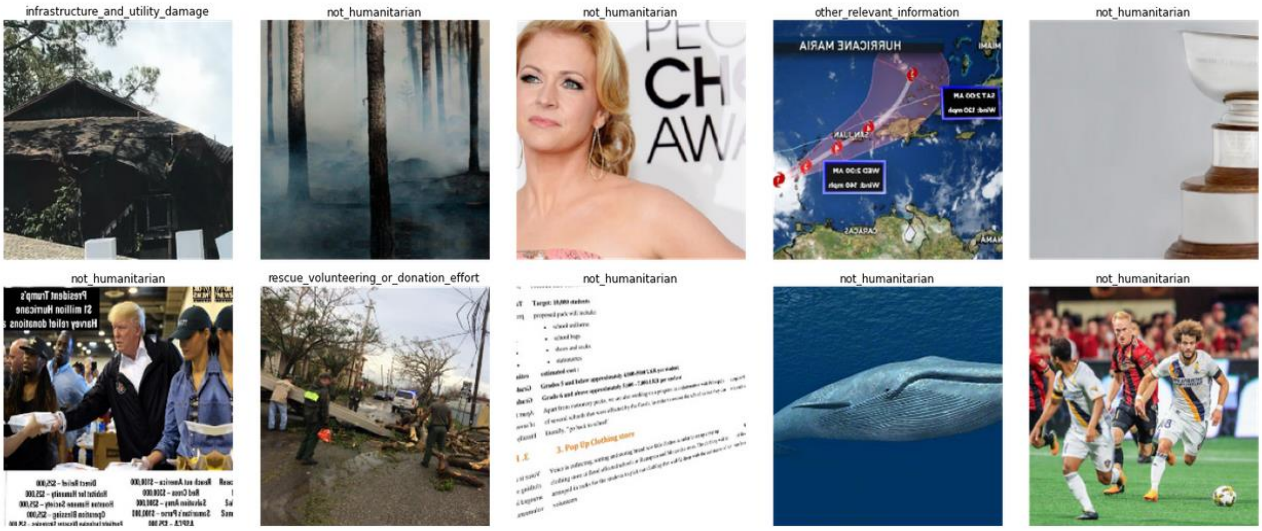


Figure3. Pre-processed Image Batch Sample

3. Model Description

I build a multimodal classification model which process data of each modality independently in all the layer's (pre-processing layers) preceding softmax layer of both pre-trained text and image models then combines the feature vectors representation of both modalities into one using these techniques, given below followed by terminal network consist of a sequence of batch normalization, dropout and a fully connected layer followed by a softmax layer. I use CrossEntropyLoss and Adam as loss function and optimizer respectively. To avoid overfitting, I use the early-stopping condition.

1. Intermediate Fusion

In intermediate fusion (feature fusion) we can take the output of an arbitrary layer of the pre-trained model before the softmax layer and the concatenate them into one, the internal representation at each stage loses some of the information contained in the representation in the previous stage but at each stage of processing data gets more separable which is easier to classify. Therefore, there exists a trade-off between the ease of classification and the information content in the representations of the network layers. The layer where the fusion of different modality features take place is called a fusion layer or a shared representation layer. Concatenating the processed output feature vectors of both models' penultimate fully connected layer in an integrated feature representation, pass it to the terminal network with output units equal to the number of classes of that task works pretty well as suggested in this paper.

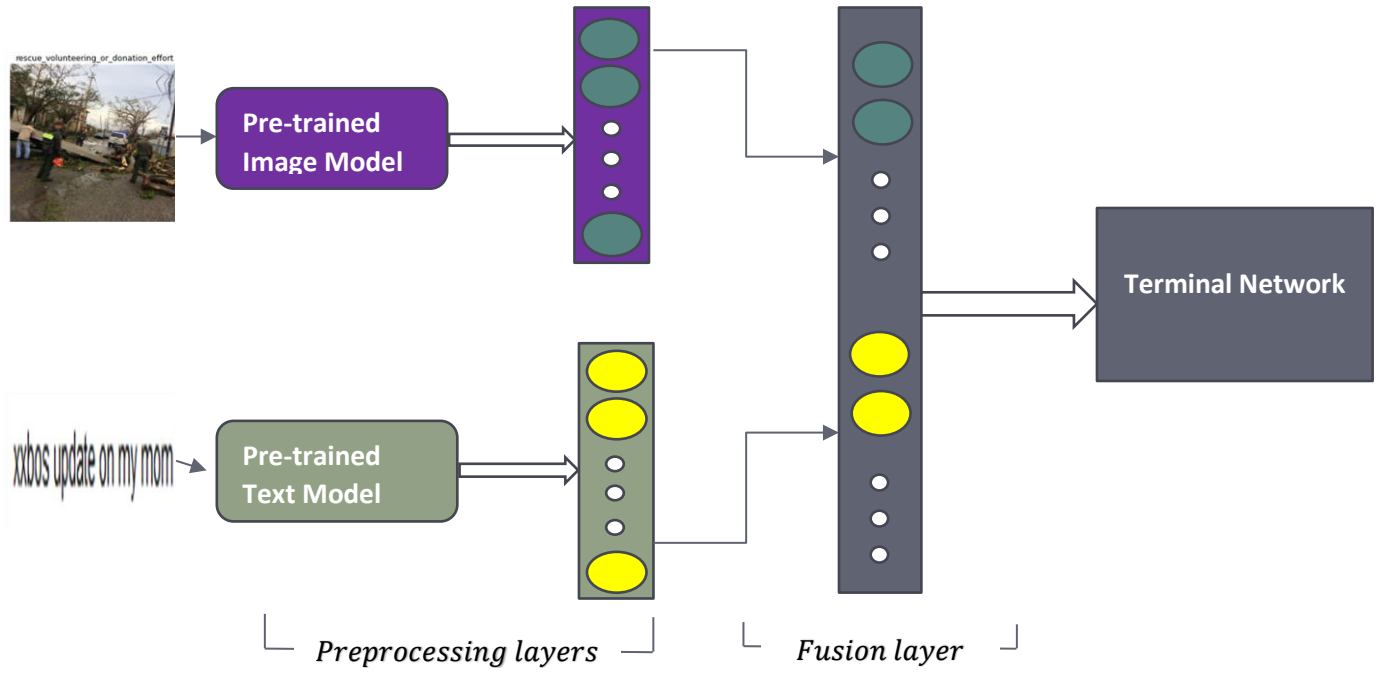


Figure4. Intermediate Fusion Model

2. EmbraceNet Fusion

In this method, I construct the model as above but instead of directly concatenating the feature vectors of text and image model, I combine the processed features into one based on the EmbraceNet architecture, and finally determines the task classes by passing it to the terminal network.

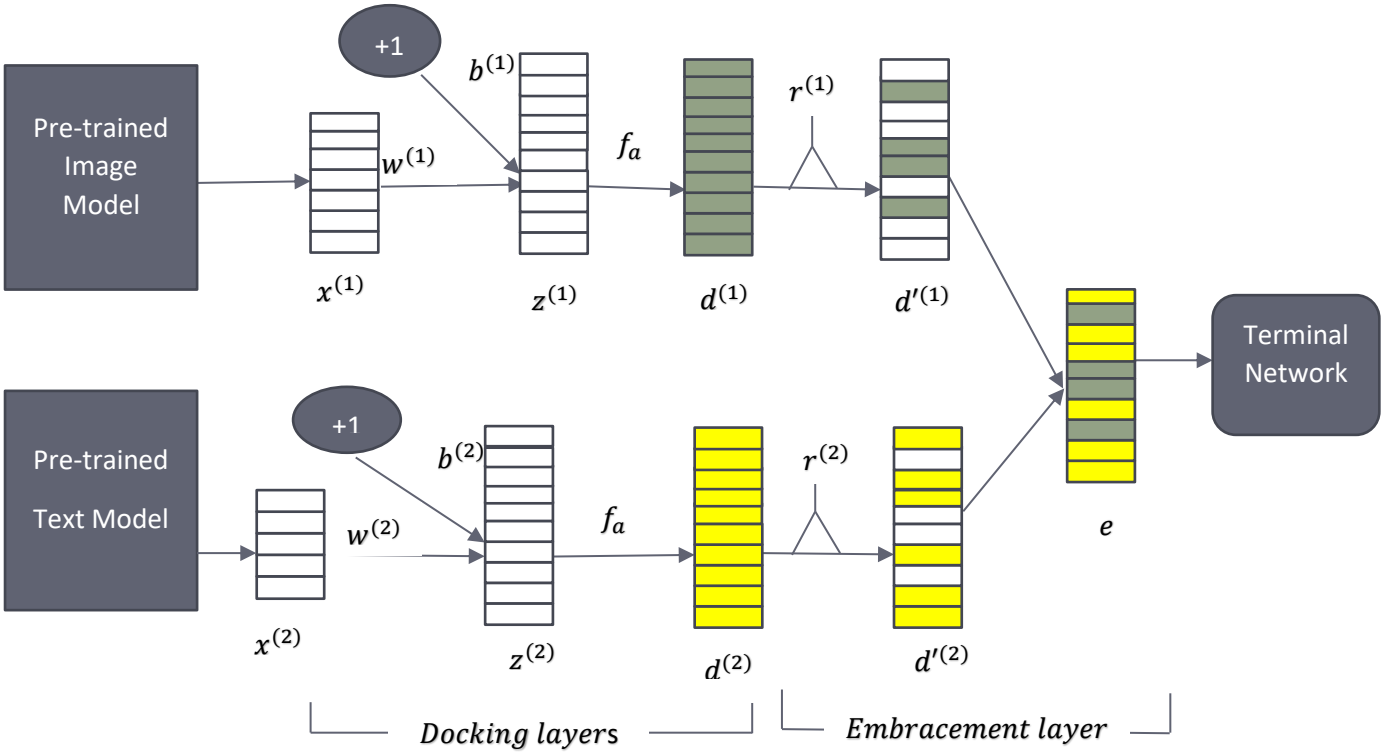


Figure5. EmbraceNet Fusion Model

Let $x^{(1)}$ and $x^{(2)}$ be the output feature vectors from both pre-trained image and text model's penultimate fully connected layer respectively.

$$z^{(k)} = w^{(k)} \cdot x^{(k)} + b^{(k)}$$
$$d^{(k)} = f_a(z^{(k)})$$

Here $w^{(k)}$ and $b^{(k)}$ are a weight vector and a bias, respectively (k=1,2). Both $d^{(k)}$ are c-dimensional vectors. An activation function f_a applied to $z^{(k)}$ to obtain the output of the kth docking layer. $r^{(k)}$ is a vector similar to $d^{(k)}$ in the dimension that is drawn from a multinomial distribution with probability p for each modality. I have used p = ½ to give an equal probability for both modalities.

$$d'^{(k)} = r^{(k)} \circ d^{(k)} \qquad \text{where } \circ \text{ elementwise product}$$

$$e = \sum_k d'^{(k)}$$

For further information you can refer the embracenet paper.

4. Experimental Setup

Various experiments have been conducted to compare the performance of different combinations of models with both fusion methods and I got a good improvement over the previous best results on these tasks. All the results are evaluated on the validation and test split.

Implementation

I implemented the multimodal model consist of several different combinations of pre-trained image and text models, one combination at a time, using the Fastai, and PyTorch frameworks in integration with Transformers. Normally the batch size of 32 is taken but for Resnet152 and Densenet161 combinations, it is 16 because of their very large number of parameters. I use a 224x224 image size. All layers of the model have been divided into 4 groups to train the model using a discriminative fine-tuning technique, for every model, after freezing all the layer groups except the last one, the learning rate is determined using learning rate finder Fastai tool and gradually unfreezing is done, finally, after unfreezing all the groups once again learning rate is determined as previously and train the model using this new learning rate. Momentum range and weight decay are taken (0.8, 0.7) and 0.1 respectively. Dropout value, 0.8 has been used for the terminal network. The rest of the hyperparameter values is taken as default values in the Fastai library.

Discriminative fine-tuning uses different learning rate for each layer groups instead of the same learning rate for all the layer groups throughout the training. As mentioned in the ULMFit paper under the discriminative fine-tune heading, I first find out learning rate using a learning rate finder tool then it is used for last layer groups and every preceding layer group's learning rate is calculated by dividing its succeeding layer group's learning rate to 2.6, which have been found empirically as mentioned in the above paper works pretty well.

I use a **gradual unfreezing** method to train the model as mentioned in same ULMFit paper under the gradual unfreezing heading, rather than fine-tuning all the layer groups at once, I unfreeze one by one starting from the last layer group to the first for fine-tuning.

5. Results

I report all the results in same the metrics as creators of this dataset does in their paper for better comparison and data creator's results can be found in this paper and here is their source code.

These metrics are Accuracy, Precision, Recall, and F1-Score for all the classes separately along with their macro and weighted (micro) average over all the classes for my best model. I also present the confusion matrix for these models. For the rest of multimodal models, I only report accuracy and the micro average of precision, recall, and f1-scores over all the classes as they report in their paper, if you want to see the detailed result of these models as well then you can see it in the source code notebook for that combination in this repository of my models.

Micro average of Accuracy, Precision, Recall, and F1-Score on test split for the models which are consist of different combinations of image and text model along with unimodal models is provided in the table given below.

Method's used to combine the models or unimodal models	Models Name	Humanitarian Task				Informativeness Task			
		Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
Intermediate	Resnet50+XLNET	0.81	0.81	0.81	0.80	0.84	0.84	0.84	0.84
Intermediate	Resnet50+XLM	0.81	0.81	0.81	0.80	0.86	0.85	0.86	0.85
Intermediate	Resnet152+XLNET	0.81	0.80	0.81	0.80	0.85	0.85	0.85	0.85
Intermediate	Densenet161+BERT	0.80	0.80	0.80	0.80	0.85	0.85	0.85	0.85
Intermediate	Resnet50+RoBERTa	0.80	0.80	0.80	0.80	0.85	0.85	0.85	0.85
Intermediate	Densenet161+DistillBERT	0.79	0.79	0.79	0.79	0.86	0.85	0.86	0.85
Intermediate	Vgg19_bn+XLM	0.81	0.81	0.81	0.81	0.85	0.84	0.85	0.85
Intermediate	Densenet161+XLNET	0.82	0.81	0.81	0.81	0.86	0.86	0.86	0.86
EmbraceNet	Densenet161+XLM	0.79	0.78	0.79	0.78	0.85	0.86	0.85	0.85
EmbraceNet	Resnet152+XLM	0.81	0.81	0.81	0.81	0.85	0.85	0.85	0.85
Intermediate	**Resnet50+AWD_LSTM**	**0.85**	**0.85**	**0.85**	**0.85**	**0.88**	**0.88**	**0.88**	**0.88**
Text only	**AWD_LSTM**	**0.75**	**0.76**	**0.75**	**0.75**	**0.84**	**0.84**	**0.84**	**0.84**
Image only	**Resnet50**	**0.81**	**0.81**	**0.81**	**0.80**	**0.85**	**0.85**	**0.85**	**0.85**
Data Creator's best Models									
	VGG16+KimCNN	**0.784**	**0.785**	**0.780**	**0.783**		**0.841**	**0.840**	**0.842**
	Text only	**0.704**	**0.700**	**0.700**	**0.677**		**0.810**	**81.0**	**0.809**
	Image only	**0.768**	**0.764**	**0.768**	**0.763**		**0.831**	**83.3**	**0.832**

Here is the detailed result of my best multimodal model (Resnet50+AWD_LSTM) along with unimodal models for both classification tasks.

Classification: Informativeness Task

Resnet50+AWD_LSTM (Image+Text)

Dev Set

	precision	recall	f1-score	support
informative	0.89	0.92	0.91	1056
not_informative	0.82	0.78	0.80	517
accuracy			0.87	1573
macro avg	0.86	0.85	0.85	1573
weighted avg	0.87	0.87	0.87	1573

Test Set

	precision	recall	f1-score	support
informative	0.91	0.92	0.91	1030
not_informative	0.83	0.81	0.82	504
accuracy			0.88	1534
macro avg	0.87	0.86	0.87	1534
weighted avg	0.88	0.88	0.88	1534

AWD_LSTM(Text only)

Dev Set

	precision	recall	f1-score	support
informative	0.84	0.92	0.88	1056
non-informative	0.79	0.65	0.72	517
accuracy			0.83	1573
macro avg	0.82	0.78	0.80	1573
weighted avg	0.83	0.83	0.83	1573

Test Set

	precision	recall	f1-score	support
informative	0.87	0.89	0.88	1030
non-informative	0.77	0.74	0.76	504
accuracy			0.84	1534
macro avg	0.82	0.82	0.82	1534
weighted avg	0.84	0.84	0.84	1534

Resnet50(Image only)

Dev Set

	precision	recall	f1-score	support
Informative	0.87	0.92	0.89	1056
Non-Informative	0.81	0.72	0.76	517
accuracy			0.85	1573
macro avg	0.84	0.82	0.83	1573
weighted avg	0.85	0.85	0.85	1573

Test Set

	precision	recall	f1-score	support
Informative	0.88	0.90	0.89	1030
Non-Informative	0.78	0.74	0.76	504
accuracy			0.85	1534
macro avg	0.83	0.82	0.82	1534
weighted avg	0.85	0.85	0.85	1534

Confusion Matrices for Informativeness Task on the Test split

Text only

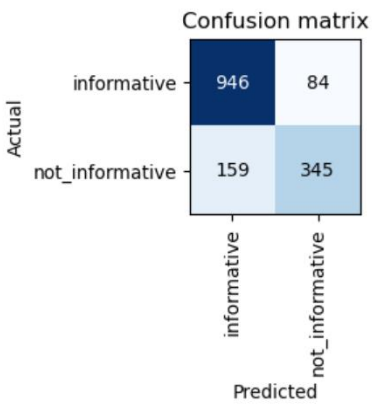
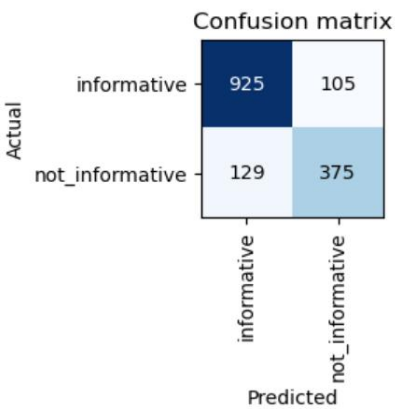
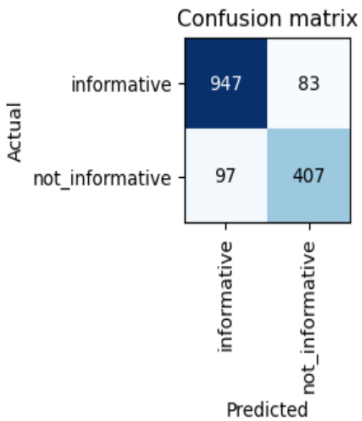


Image only



Image+Text



Data creator's confusion matrices for informativeness task on test split

(a) Text-only

		Predicted Inf	*Predicted* Not-inf
Human	Inf	875	155
	Not-inf	139	365

(b) Image-only

		Predicted Inf	*Predicted* Not-inf
Human	Inf	916	114
	Not-inf	145	359

(c) Text + Image

		Predicted Inf	*Predicted* Not-inf
Human	Inf	929	101
	Not-inf	139	365

Classification: Humanitarian Task

Resnet50+AWD_LSTM (Image+Text)

Dev Set

	precision	recall	f1-score	support
affected_individuals	0.00	0.00	0.00	9
infrastructure_and_utility_damage	0.73	0.82	0.77	80
not_humanitarian	0.86	0.88	0.87	521
other_relevant_information	0.84	0.80	0.82	239
rescue_volunteering_or_donation_effort	0.77	0.75	0.76	149
accuracy			0.83	998
macro avg	0.64	0.65	0.64	998
weighted avg	0.82	0.83	0.83	998

Test Set

	precision	recall	f1-score	support
affected_individuals	1.00	0.44	0.62	9
infrastructure_and_utility_damage	0.81	0.88	0.84	81
not_humanitarian	0.87	0.90	0.89	504
other_relevant_information	0.87	0.83	0.85	235
rescue_volunteering_or_donation_effort	0.76	0.71	0.73	126
accuracy			0.85	955
macro avg	0.86	0.75	0.78	955
weighted avg	0.85	0.85	0.85	955

AWD_LSTM(Text only)

Valid Set

	precision	recall	f1-score	support
affected_individuals	1.00	0.22	0.36	9
infrastructure_and_utility_damage	0.73	0.66	0.69	80
not_humanitarian	0.81	0.88	0.84	521
other_relevant_information	0.74	0.71	0.72	239
rescue_volunteering_or_donation_effort	0.79	0.68	0.73	149
accuracy			0.78	998
macro avg	0.81	0.63	0.67	998
weighted avg	0.78	0.78	0.78	998

Test Set

	precision	recall	f1-score	support
affected_individuals	1.00	0.11	0.20	9
infrastructure_and_utility_damage	0.74	0.60	0.67	81
not_humanitarian	0.76	0.88	0.82	504
other_relevant_information	0.78	0.62	0.69	235
rescue_volunteering_or_donation_effort	0.70	0.63	0.67	126
accuracy			0.75	955
macro avg	0.80	0.57	0.61	955
weighted avg	0.76	0.75	0.75	955

Resnet50(Image only)

Valid

	precision	recall	f1-score	support
affected_individuals	0.20	0.11	0.14	9
infrastructure_and_utility_damage	0.62	0.76	0.69	80
not_humanitarian	0.80	0.88	0.84	521
other_relevant_information	0.83	0.70	0.76	239
rescue_volunteering_or_donation_effort	0.77	0.63	0.69	149
accuracy			0.78	998
macro avg	0.64	0.62	0.62	998
weighted avg	0.78	0.78	0.78	998

Test

	precision	recall	f1-score	support
affected_individuals	0.60	0.33	0.43	9
infrastructure_and_utility_damage	0.73	0.78	0.75	81
not_humanitarian	0.81	0.91	0.85	504
other_relevant_information	0.89	0.77	0.83	235
rescue_volunteering_or_donation_effort	0.72	0.53	0.61	126
accuracy			0.81	955
macro avg	0.75	0.66	0.70	955
weighted avg	0.81	0.81	0.80	955

Confusion Matrices for Humanitarian Task on the Test split

Text only

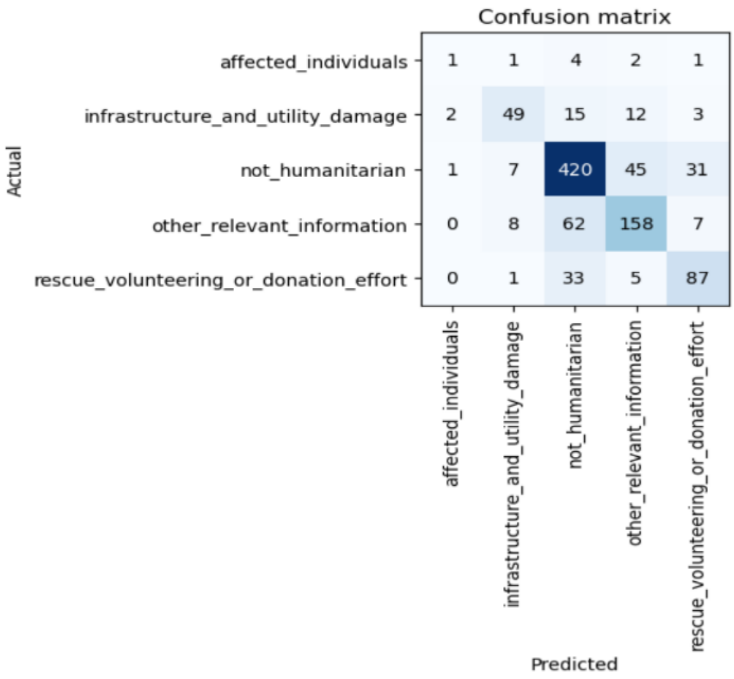
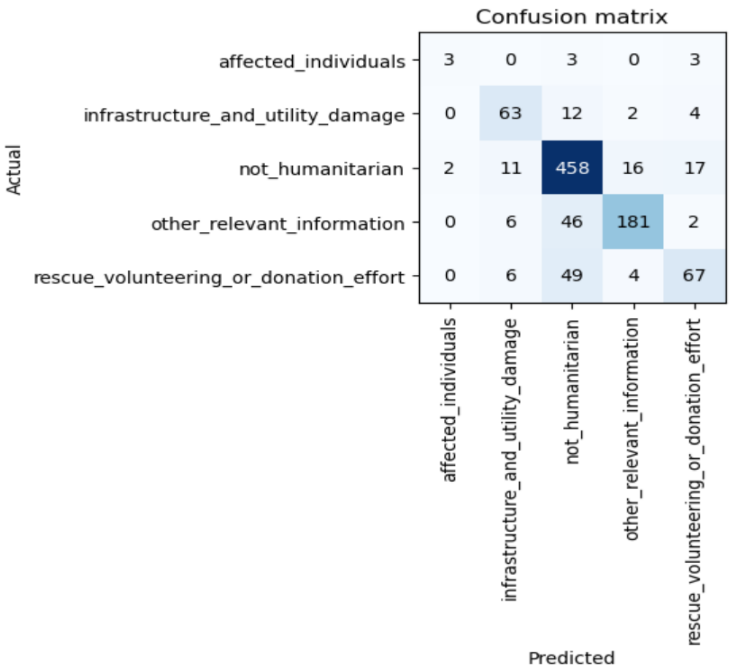
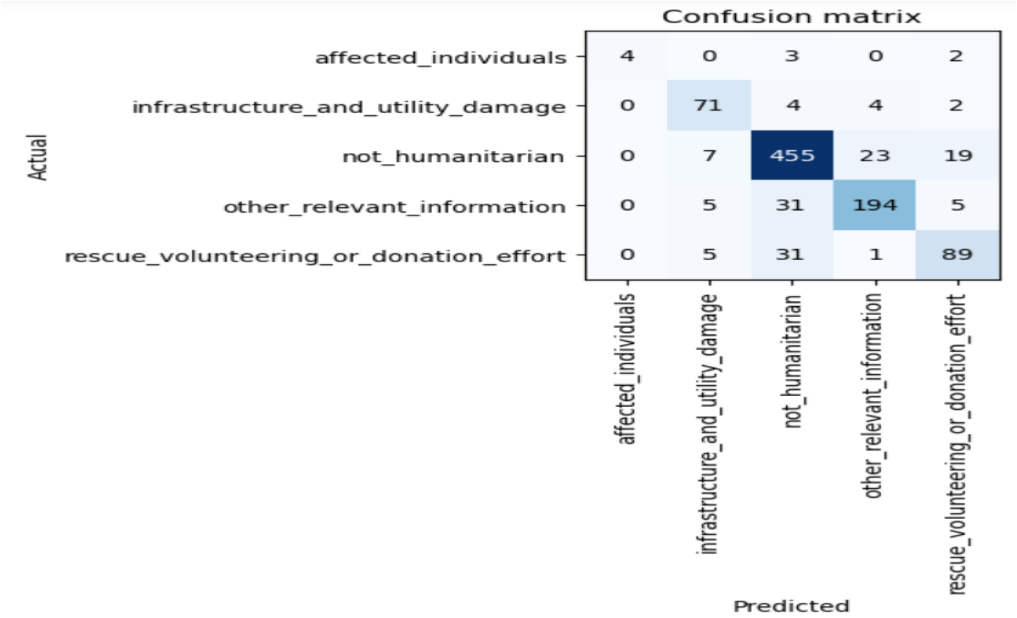


Image only



Image+Text



Data creator's confusion matrices for humanitarian task on the test split

(a) Text-only

		Predicted				
		A	I	N	O	R
Human	A	0	0	5	1	3
	I	0	17	41	12	11
	N	0	1	458	20	25
	O	0	6	105	112	12
	R	0	2	37	2	85

(b) Image-only

		Predicted				
		A	I	N	O	R
Human	A	1	0	4	0	4
	I	1	56	13	6	5
	N	0	13	437	22	32
	O	0	5	50	178	2
	R	0	5	43	5	73

(c) Text + Image

		Predicted				
		A	I	N	O	R
Human	A	1	0	3	0	5
	I	1	61	10	4	5
	N	0	17	426	26	35
	O	0	3	49	180	3
	R	0	9	33	3	81

It can be observed from the above unimodal experiments' result, the image-only models perform better than the text-only models by a margin of more than 1% and 5% in both informativeness and humanitarian tasks respectively. The multimodal model additionally performs better than the image model with an improvement of more than 3% and 4% on both the informativeness and humanitarian tasks. As the complexity of the humanitarian task is more because of the greater number of classes so the overall performance of the informativeness task is better than the humanitarian task. These results suggest the multimodal information helps the model to perform better than using unimodal information only.

6. Conclusion

Identifying damage and human casualties in real-time from social media posts is critical for providing prompt and suitable resources and medical attention, to save as many lives as possible. Although the information on social media can be of different modalities such as texts, images, audio, or videos, traditional approaches in classification usually leverage only one prominent modality. In this internship, my objective was to improve the classification performance over the previous best on the CrisisMMD multimodal dataset, consisting of image and text modality, by building multimodal deep learning architecture utilizing both image and text data, which could be helpful to the humanitarian aid organizations to address the above issue or any further study. My modal employed pre-trained text and image model's pre-processing layers to process the data of both modalities independently and concatenate the learned feature vectors into an integrated representation using intermediate or embracenet fusion method. I conducted several experiments on this multimodal dataset with the many combinations of different pre-trained image and text models using transfer learning out which Resnet50+AWD_LSTM with intermediate fusion is the best performer. I got pretty well improvement over the earlier best result on this dataset for both the task.