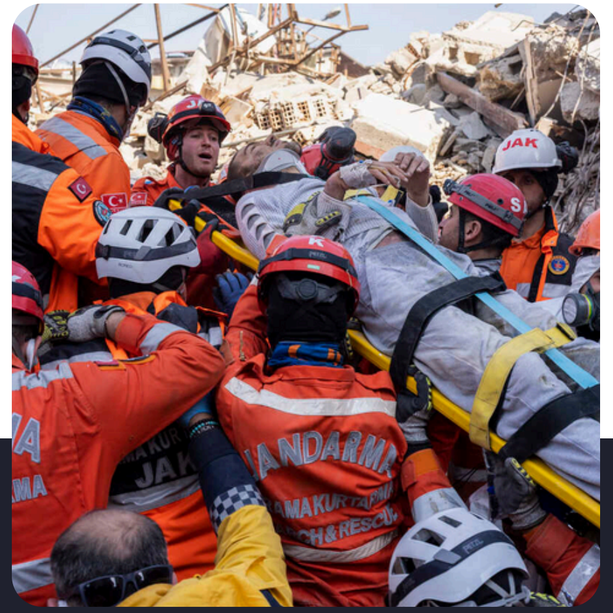
AI FOR CRISIS CLARITY



WHEN WORDS AND IMAGES SPEAK, AI LISTENS

Multimodal Disaster Tweet Classification with VisualBERT

PRESENTED BY TEAM I:

- RAPHAEL VINCENT GABRIEL DATO
- HANG DONG
- YOUSSEF KANDIL

WHAT WOULD YOU DO TO SURVIVE?

You’re now trapped beneath a collapsed concrete pillar. You’re injured, bleeding, and can’t move. Your phone has 20% battery, and barely two signal bars.

A. Call emergency services
B. Try to remove the heavy debris yourself
C. Search nearby for tools or a first-aid kit
D. Shout for help continuously
E. Post a geo-tagged tweet with a photo of your location

A. CALL EMERGENCY HOTLINE

Phone lines are often jammed during mass disasters; even if it connects, locating you takes time

ESTIMATED SURVIVAL PROBABILITY: 12%

Note: These survival probabilities are estimated based on global disaster reports and real-world rescue patterns.

B. TRY TO REMOVE DEBRIS YOURSELF

You’re severely injured; without proper tools and others’ help, moving debris alone can worsen your condition

ESTIMATED SURVIVAL PROBABILITY: 8%

Note: These survival probabilities are estimated based on global disaster reports and real-world rescue patterns.

C. LOOK FOR NEARBY TOOLS OR MEDICINE

You’re pinned and unable to move; nothing useful is within reach in your immediate surroundings

ESTIMATED SURVIVAL PROBABILITY: 5%

Note: These survival probabilities are estimated based on global disaster reports and real-world rescue patterns.

D. SHOUT CONTINUOUSLY FOR HELP

Sound rarely travels far under rubble; prolonged shouting drains your strength and accelerates dehydration

ESTIMATED SURVIVAL PROBABILITY: 10%

Note: These survival probabilities are estimated based on global disaster reports and real-world rescue patterns.

E. POST A GEO-TAGGED TWEET WITH A PHOTO

Emergency responders now use AI to scan social media; combining image + text + GPS dramatically boosts visibility

ESTIMATED SURVIVAL PROBABILITY: 61%

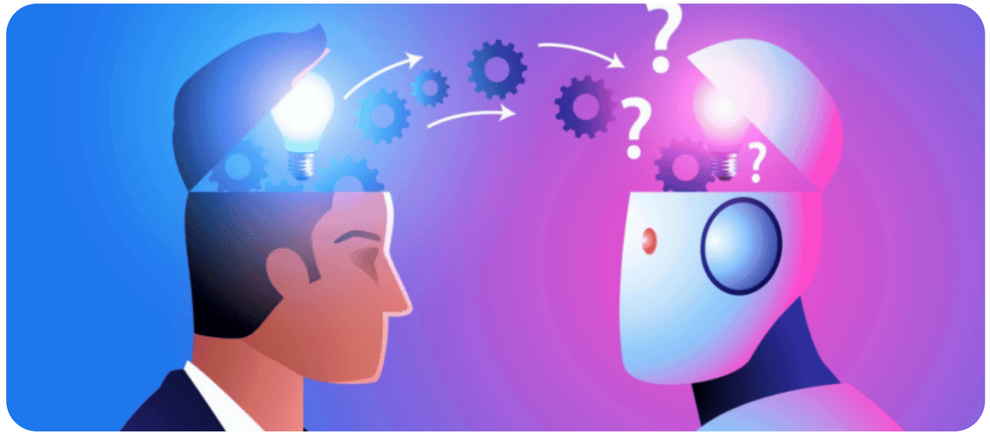
Note: These survival probabilities are estimated based on global disaster reports and real-world rescue patterns.

WHAT PROBLEM ARE WE SOLVING?

- Social media platforms contain valuable real-time information
- Prior works use shallow fusion, lacking fine-grained semantic alignment

We use VisualBERT, bringing **deeper semantic alignment** between the two modalities

WHY DOES IT MATTER?



SOCIAL IMPACT

- SDG 3 - Good Health and Well-being
- SDG 9 – Industry, Innovation and Infrastructure
- SDG 11 – Sustainable Cities and Communities
- SDG 16 – Peace, Justice and Strong Institutions

TECHNICAL GAP

- Real-world crisis data = messy & noisy
- We test how joint models adapt to these settings

DATASET OVERVIEW

Source: CrisisMMD (Hugging Face)

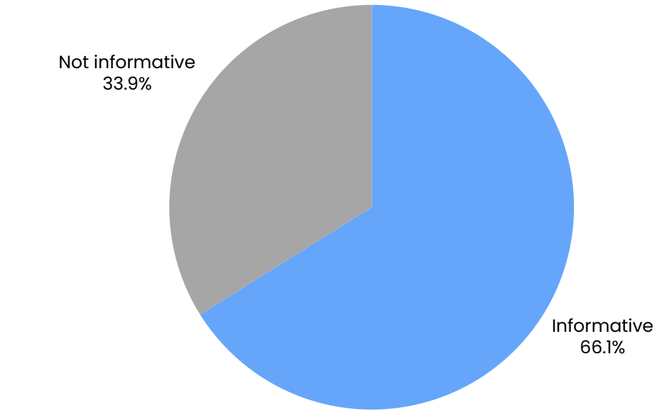
Modalities: Text, Image, Label

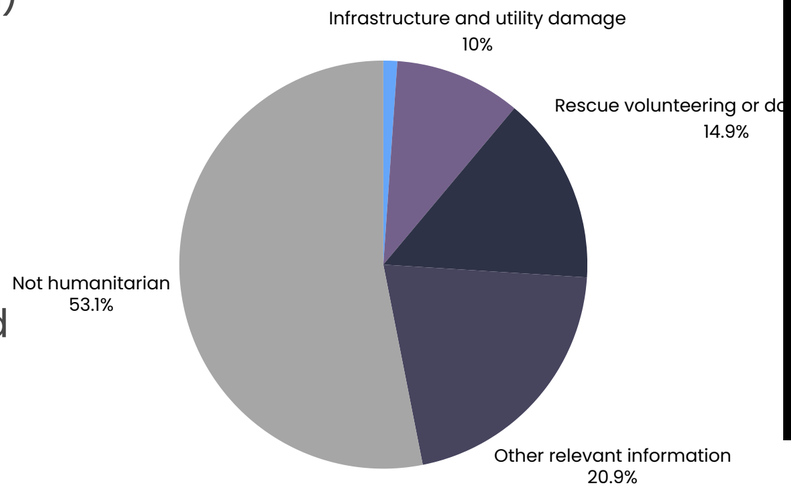
Tasks:

- Task 1: Informative vs. Not (Binary)
- Task 2: Humanitarian Categories (Multi-label)

Challenges:

- Cross-modal data
- Imbalanced label distribution
- Some redundancy and noise in both text and images





#5438

event_name

california_wildfires

tweet_id

920,352,459,297,558,500

image_id

920352459297558529_0

tweet_text

California wildfire hero dodged debris with disabledÂ roommate https://t.co/9rFTxP3oXv https://t.co/3NJmwfhc2o

image

label

affected_individuals

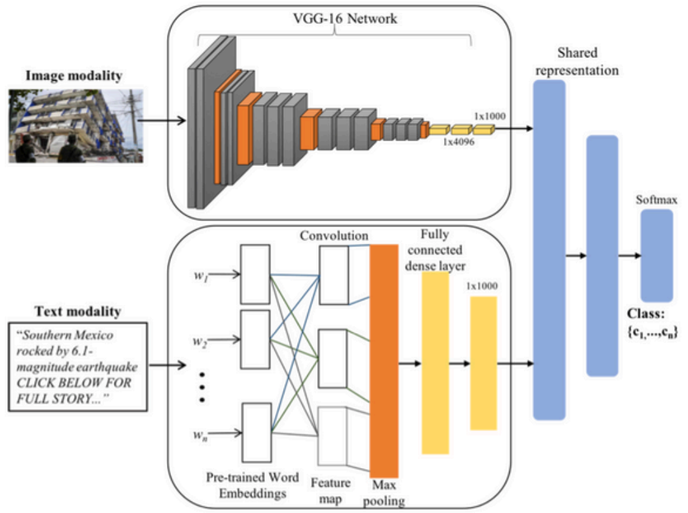
PREVIOUS WORK

1. Ofli et al. (2020)
   - CNN-based model for text and image feature

Training mode	Modality	Accuracy	Precision	Recall	F1-score
Unimodal	Text	80.8	81.0	81.0	80.9
	Image	83.3	83.1	83.3	83.2
Multimodal	Text + Image	84.4	84.1	84.0	84.2

Table 3. Results for the humanitarian classification task.

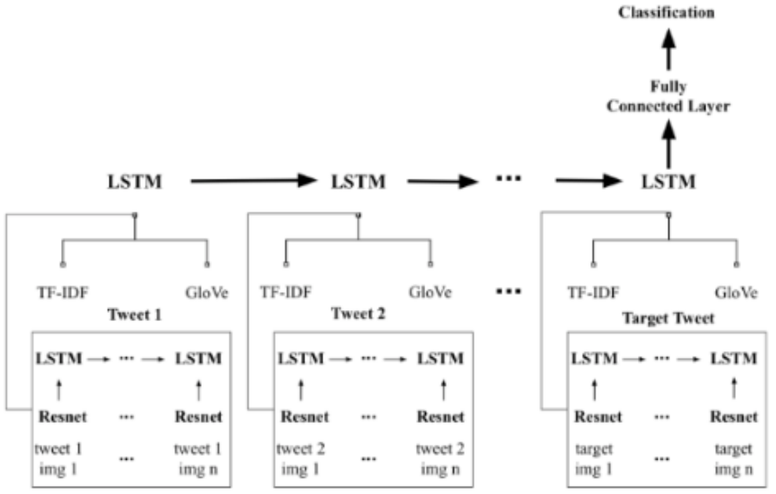
Training mode	Modality	Accuracy	Precision	Recall	F1-score
Unimodal	Text	70.4	70.0	70.0	67.7
	Image	76.8	76.4	76.8	76.3
Multimodal	Text + Image	78.4	78.5	78.0	78.3



2. IITUDND model (2020)
   - TF-IDF/GloVe for text
   - ResNet for images, and LSTM/FFNNs

Resize

Event	Accuracy	AUC	Precision	Recall	F1
Harvey - Multi Modal - Nalluru et al.	0.874	0.9065			
Harvey - Multi Modal - LSTM	0.8799	0.8757	0.8988	0.9601	0.9284
Harvey - Multi Modal - With Histories	0.8681	0.8571	0.8946	0.9493	0.9211
Harvey - Multi Modal - Just Labeled	0.8781	0.8914	0.9047	0.9499	0.9267
Harvey - Text - Nalluru et al.	0.869	0.900			
Harvey - Text - LSTM	0.8761	0.8629	0.8855	0.9731	0.9273
Harvey - Text - With Histories	0.8730	0.8705	0.8933	0.9579	0.9245
Harvey - Text - Just Labeled	0.8666	0.8718	0.8883	0.9559	0.9208

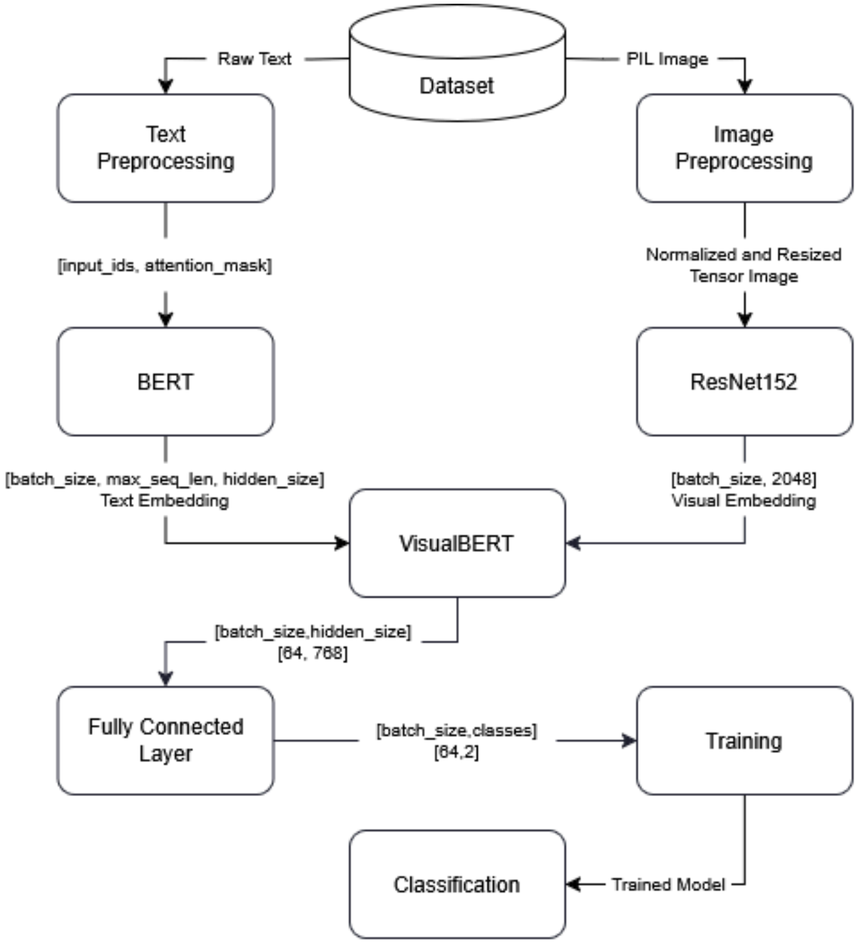


METHODOLOGY

BERT — RESNET 152 — VISUALBERT

Hyperparameters used:

- BERT Tokenization
  - Maximum Sequence Length: 128
- VisualBERT
  - Dropout Rate: 0.3
- Fully Connected Layers
  - Batch Size: 64
  - Learning Rate: 1e-5
  - Weight Decay: 1e-4
  - Number of Workers: 4
  - Patience: 2
  - Factor: 0.5
- Training
  - Epochs: 20



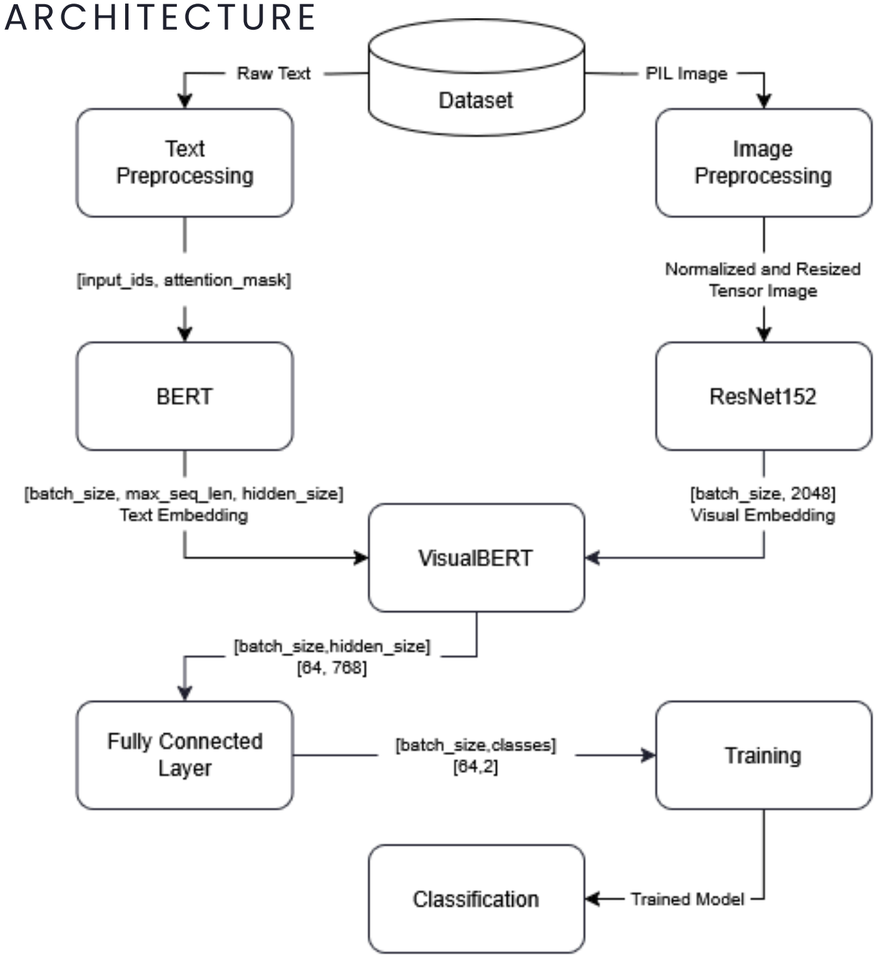
METHODOLOGY

BERT — ???? — VISUALBERT

Visual Embedding Experiments and Selection:

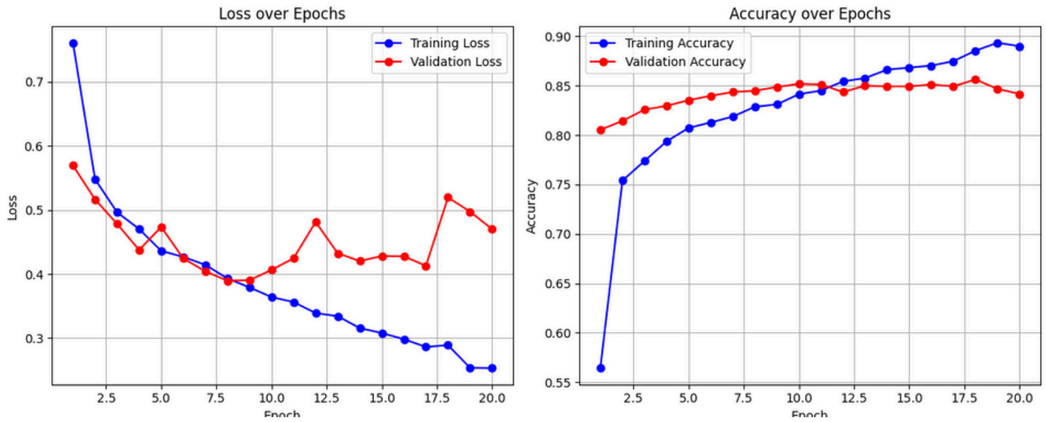
- Resnet 50 ✅
- Resnet 152 ✅
- Detectron2 (Meta 2019) ❌
- DINOv2 (Meta 2023) ✅

ARCHITECTURE



METHODOLOGY

Resnet 50



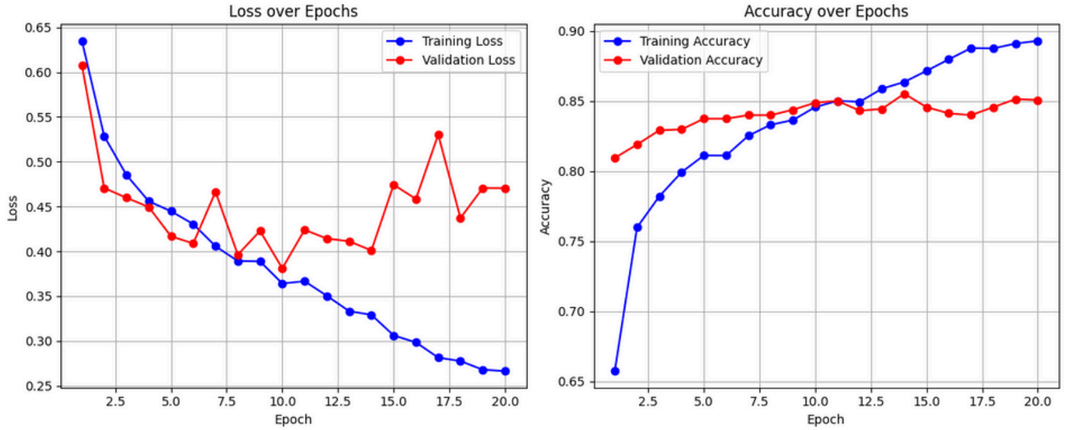
Validation Loss: 0.4701
Validation Report:

	precision	recall	f1-score	support
Uninformative	0.75	0.78	0.76	517
Informative	0.89	0.88	0.88	1056
accuracy			0.84	1573
macro avg	0.82	0.83	0.82	1573
weighted avg	0.84	0.84	0.84	1573

Test Loss: 0.4839
Test Report:

	precision	recall	f1-score	support
Uninformative	0.75	0.79	0.77	504
Informative	0.89	0.87	0.88	1030
accuracy			0.84	1534
macro avg	0.82	0.83	0.83	1534
weighted avg	0.85	0.84	0.85	1534

Resnet 152



Validation Loss: 0.4707
Validation Report:

	precision	recall	f1-score	support
Uninformative	0.79	0.74	0.76	517
Informative	0.88	0.91	0.89	1056
accuracy			0.85	1573
macro avg	0.83	0.82	0.83	1573
weighted avg	0.85	0.85	0.85	1573

Test Loss: 0.4859
Test Report:

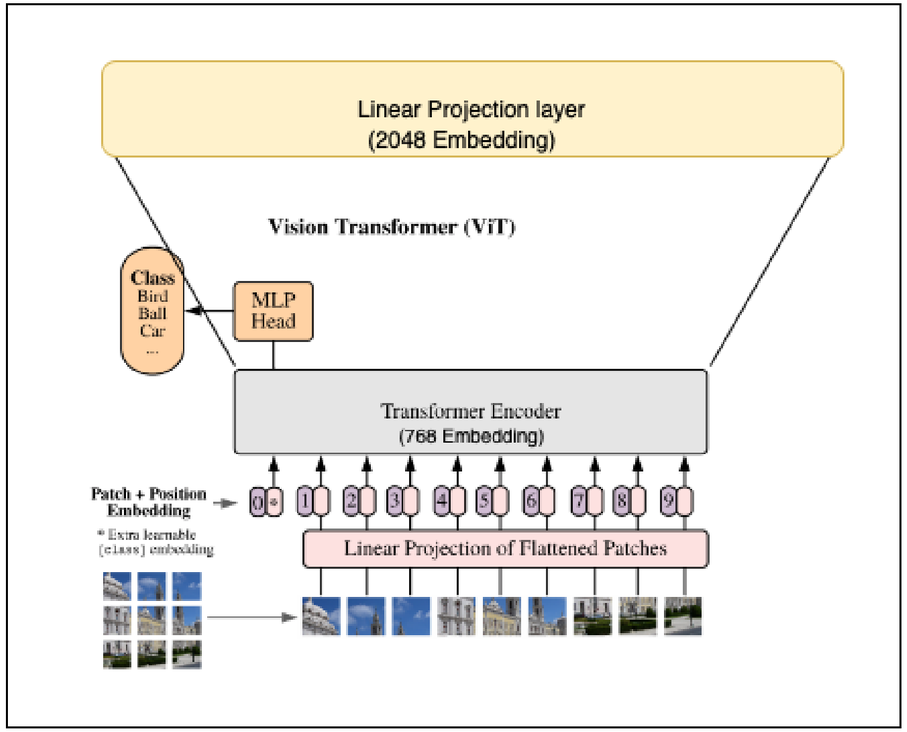
	precision	recall	f1-score	support
Uninformative	0.79	0.76	0.78	504
Informative	0.88	0.90	0.89	1030
accuracy			0.86	1534
macro avg	0.84	0.83	0.83	1534
weighted avg	0.85	0.86	0.86	1534

SOLUTION AND METHODOLOGY

BERT — ???? — VISUALBERT

- DINOv2 produces a visual embedding of size 768
- Visualbert expects a visual embedding of size 2048
- Introduced a “linear projection layer”

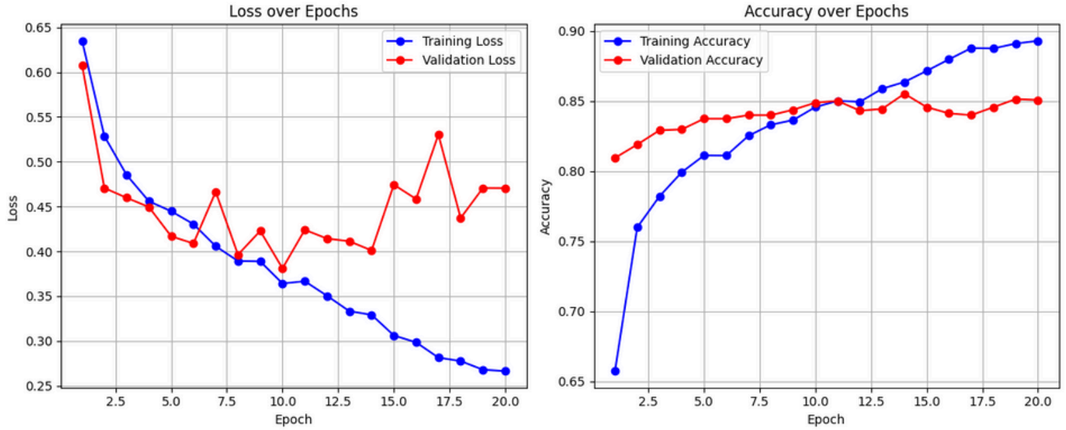
DINOv2 Projection Architecture



https://paperswithcode.com/method/vision-transformer

SOLUTION AND METHODOLOGY

Resnet 152



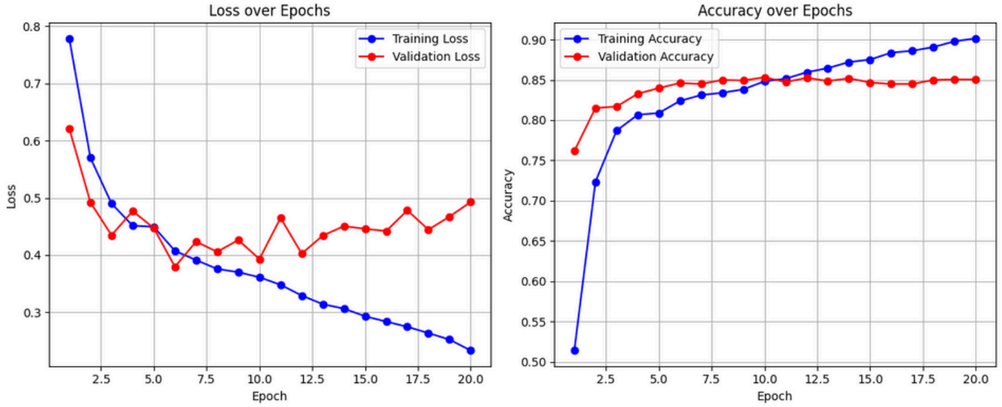
Validation Loss: 0.4707
Validation Report:

	precision	recall	f1-score	support
Uninformative	0.79	0.74	0.76	517
Informative	0.88	0.91	0.89	1056
accuracy			0.85	1573
macro avg	0.83	0.82	0.83	1573
weighted avg	0.85	0.85	0.85	1573

Test Loss: 0.4859
Test Report:

	precision	recall	f1-score	support
Uninformative	0.79	0.76	0.78	504
Informative	0.88	0.90	0.89	1030
accuracy			0.86	1534
macro avg	0.84	0.83	0.83	1534
weighted avg	0.85	0.86	0.86	1534

DINOv2



Validation Loss: 0.4927
Validation Report:

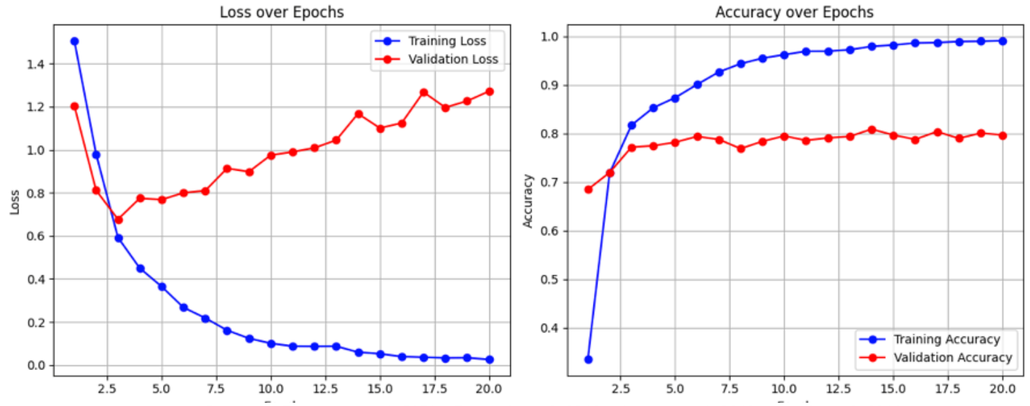
	precision	recall	f1-score	support
Uninformative	0.80	0.72	0.76	517
Informative	0.87	0.91	0.89	1056
accuracy			0.85	1573
macro avg	0.84	0.82	0.83	1573
weighted avg	0.85	0.85	0.85	1573

Test Loss: 0.5212
Test Report:

	precision	recall	f1-score	support
Uninformative	0.83	0.71	0.77	504
Informative	0.87	0.93	0.90	1030
accuracy			0.86	1534
macro avg	0.85	0.82	0.83	1534
weighted avg	0.85	0.86	0.85	1534

Results - Humanitarian

ACCURACY — LOSS — PRECISION — RECALL — F1-SCORE




```
Epoch 1, Train Loss: 1.5071, Train Acc: 0.3346, Val Loss: 1.2049, Val Acc: 0.6844
Epoch 2, Train Loss: 0.9796, Train Acc: 0.7186, Val Loss: 0.8112, Val Acc: 0.7194
Epoch 3, Train Loss: 0.5925, Train Acc: 0.8168, Val Loss: 0.6770, Val Acc: 0.7715
Epoch 4, Train Loss: 0.4498, Train Acc: 0.8528, Val Loss: 0.7746, Val Acc: 0.7745
Epoch 5, Train Loss: 0.3646, Train Acc: 0.8733, Val Loss: 0.7681, Val Acc: 0.7816
Epoch 6, Train Loss: 0.2673, Train Acc: 0.9004, Val Loss: 0.8000, Val Acc: 0.7936
Epoch 7, Train Loss: 0.2182, Train Acc: 0.9262, Val Loss: 0.8098, Val Acc: 0.7876
Epoch 8, Train Loss: 0.1612, Train Acc: 0.9434, Val Loss: 0.9137, Val Acc: 0.7685
Epoch 9, Train Loss: 0.1237, Train Acc: 0.9546, Val Loss: 0.8979, Val Acc: 0.7836
Epoch 10, Train Loss: 0.1007, Train Acc: 0.9620, Val Loss: 0.9748, Val Acc: 0.7946
Epoch 11, Train Loss: 0.0871, Train Acc: 0.9690, Val Loss: 0.9899, Val Acc: 0.7856
Epoch 12, Train Loss: 0.0863, Train Acc: 0.9691, Val Loss: 1.0086, Val Acc: 0.7906
Epoch 13, Train Loss: 0.0870, Train Acc: 0.9721, Val Loss: 1.0447, Val Acc: 0.7936
Epoch 14, Train Loss: 0.0595, Train Acc: 0.9789, Val Loss: 1.1678, Val Acc: 0.8086
Epoch 15, Train Loss: 0.0521, Train Acc: 0.9819, Val Loss: 1.1009, Val Acc: 0.7966
Epoch 16, Train Loss: 0.0389, Train Acc: 0.9860, Val Loss: 1.1241, Val Acc: 0.7876
Epoch 17, Train Loss: 0.0361, Train Acc: 0.9868, Val Loss: 1.2678, Val Acc: 0.8036
Epoch 18, Train Loss: 0.0328, Train Acc: 0.9891, Val Loss: 1.1965, Val Acc: 0.7896
Epoch 19, Train Loss: 0.0336, Train Acc: 0.9896, Val Loss: 1.2266, Val Acc: 0.8006
Epoch 20, Train Loss: 0.0256, Train Acc: 0.9909, Val Loss: 1.2715, Val Acc: 0.7966
```

	precision	recall	f1-score	support
affected_individuals	0.50	0.56	0.53	9
infrastructure_and_utility_damage	0.71	0.73	0.72	81
not_humanitarian	0.86	0.82	0.84	504
other_relevant_information	0.76	0.72	0.74	235
rescue_volunteering_or_donation_effort	0.71	0.87	0.79	126
accuracy			0.79	955
macro avg	0.71	0.74	0.72	955
weighted avg	0.80	0.79	0.80	955

CHALLENGES AND LIMITATIONS

DATA LIMITATIONS

Class Imbalance: Unequal Distribution of classes.

Label Noise: Annotations may be subjective or inconsistent, affecting model generalization.

COMPUTATIONAL CHALLENGES

High Memory Usage: BERT+ ResNet152 + VisualBERT + batch size of 64 consumed a lot of GPU memory

Slow Training: Fine-tuning large transformer models with image features resulted in long training times

TIME / RESOURCE CONSTRAINTS

Limited access to high-end GPUs restricted experimentation with alternative architectures and certain hyperparameters

WHAT WE WOULD DO DIFFERENTLY

Data Augmentation: Apply more advanced augmentation techniques to address the class imbalance

Deeper Hyperparameter Tuning: Use more fine-tuning tools and techniques like Grid Search to further enhance results

OTHER EXPLORATIONS

- Used a different Dataset from the one in Huggingface
- Experimented with Different Image Feature Extractors like DINO and ResNet152
- Experimented with Different Hyperparameters

RECOMMENDED NEXT STEPS

- To Use and Fine-Tune with the Dino Model
- Try a Different Multimodal Models like CLIP
- Use techniques like Grid Search to properly find the best hyperparameters

OTHER EXPLORATIONS

- Used a different Dataset from the one in Huggingface
- Experimented with Different Image Feature Extractors like DINO and ResNet152
- Experimented with Different Hyperparameters

RECOMMENDED NEXT STEPS

- To Use and Fine-Tune with the Dino Model
- Try a Different Multimodal Models like CLIP
- Use techniques like Grid Search to properly find the best hyperparameters

THANK YOU