CIS 522 – Final Project – Technical Report

Team Detectors

April 2022

Team Members:

- Akriti Gupta; akritig; Email: akritig@seas.upenn.edu
- Bhumika Singhal; bhsingha; Email: bhsingha@seas.upenn.edu
- Namita Shukla; nashukla; Email: nashukla@seas.upenn.edu

Abstract

Hateful meme detection is a well-known research area that requires both visual and linguistic understanding. It matters because in today's world information and opinions stem from multimedia. With people smartly disguising hateful intent behind apparently harmless images/text which when combined within cultural and societal context can hurt sentiments of various minority groups. Thus, there is a dire need to be able to detect such hateful multimedia in a multimodal setting.

For this purpose, we have used Facebook's hate meme detection data set specially annotated such that the unimodal priors are bound to fail, that is, the images and text individually don't hold much signal. We have used ResNext and RoBERTa unimodal models as the baselines. In order to explore the multimodality of the dataset, we used the early fusion approach by concatenating the ResNext embeddings of pure images (2047 dimensional) and RoBERTa embeddings of text (768 dimensional) and then subsequently performing classification using various fine-tuned models such as Shallow Feed Forward Network, Deep Feed Forward Network, CatBoost, LGBM, XGBoost and Logistic Regression.

Our initial analysis suggested that despite the fact that the visual embeddings were almost 2.6 times larger than the textual ones, the information they carried was not sufficient. Hence, we condensed the dimensionality of both text and image embeddings using PCA in order to capture features delivering maximum variance/signal and also experimented with the different dimensions of these embeddings to test our classification models' performance. In order to extract more signal from the textual embeddings, we fine-tuned the last 4 layers of RoBERTa pre-trained on twitter's tweets with another hate speech centric dataset from UC Berkeley to further tune the embeddings for our specific problem statement.

Our performance metric is AUC because of the skewness in the dataset. It is observed that retrained RoBERTa performs the best with a AUC of 0.66 when more weightage is given to the text embeddings (150 dimensional) over the visual embeddings (50 dimensional) after applying PCA.

1 Introduction

1.1 Problem Statement

The problem we are aiming to solve is the detection of hateful emotion in media which is needs to be derived from the combined effect of text and images as individually both the text and images can be benign however the combined intent can be hateful. In order for Deep Learning to become a more robust and reliable tool for the detection of hate speech, it must mimic humans i.e. while glancing over memes, we don't think about the words and photos independently; we understand the combined meaning together. This is extremely challenging for machines, however, because it means they can't just analyze the text and the image separately. They must combine these different modalities and understand how the meaning changes when they are presented together.

1.2 Motivation and why it matters

We all would have come across many memes in our social media feeds which can be informative, funny, hateful, or even meaningless. However, memes are now the most innovative way to spread and seed ideas into society and have evolved into an effective means of disseminating hate on the internet. The widespread use of social media has fueled hate speech or communication that conveys prejudiced messages against the members of minority groups such as Women, LGBT and Black Community. Recently, Meta and Reddit groups: r/The_Donald and 4chan have been responsible for spreading over 600,000 racist memes in a short span of 13 months [1]. Also, even though this problem statement targets the memes found on social media but this successful detection can be extended to other domains which work with multimedia i.e where meaning is derived from combination of text and images.

1.3 Big Problem in Science: Strong unimodal priors

In our case, it has been pointed out through experiments that language can inadvertently impose strong priors that result in a seemingly impressive performance, without any understanding of the visual content in the underlying model. A lot of the current models exploit this to get higher accuracies. However, this is not the complete picture in the current case as we will show through this dataset how benign text can be misleading.

(a) (b) (c)

Figure 1: Examples where benign text is non hateful but the meme is hateful

1.4 Uniqueness of this challenge and the narrower problem we are trying to solve

The Hateful Meme Challenge dataset proposed by a team of annotators at Meta AI [4] formulated using a scientific process such that the detection of hateful memes depends on benign confounders. This forces the models not to bias towards either text or image signals but focus on both. Hence, the open question is how to best feed in the information from both images and text to our AI systems so that they classify the memes as hateful or non-hateful correctly.

In addition, even though multimodality has varied applications [2,3], however, it is not evident to what extent multimodal reasoning is required for solving many current tasks and datasets.

1.5 Contribution to the field and how we close the gap

Most of the state-of-the-art methods use high computation and high dimensional embeddings. However, we aim to introduce a methodology that is flexible and lightweight in a way that solves the problem efficiently in shorter time using lesser computational resources. We observed that while image encoding models produce higher dimensional embeddings, they do not contain much signal. Hence, we used PCA to adjust the dimensionality of the embeddings, to be able to pick the features capturing maximum information and high variance. We even fine tuned RoBERTa to counter the loss of information in the low dimensional embeddings.

1.6 Our Approach

An early fusion multi-modal approach was used to solve the problem involving concatenation of "pure" images and text embeddings followed by classification using various fine-tuned models such as Shallow Feed Forward Network, Deep Feed Forward Network, CatBoost and Logistic Regression to name a few. Next, in order to get more concentrated signal from the images and text , we performed PCA on the image and text embeddings while also varying the individual embedding size. Furthermore, since unimodal RoBERTa performed better than unimodal ResNext-50 indicating that text contains more valid signal and information than images, hence we fine-tuned the last 4 layers of RoBERTa with another data set from UC Berkeley containing hate speech to further generate embeddings specific to our problem statement.

1.7 Our Results

We fed both original and PCA based embeddings to our classifiers of Shallow FFN, Deep FFN, Logistic Regression, CatBoost, XGBoost, and LGBM. We analysed the results and found that even though the models performed faster on PCA embeddings, they didn't outperform the results that the original embeddings gave us. To combat this, we fine tuned RoBERTa using UCB DLab's hate speech data to make our embeddings more specific to the classification task at hand. This gave us the best AUC of 0.66 with a Shallow FFN (2 layers: BatchNormalized and Dropout).

2 Related Work

2.1 Previous work

In most previous works, hate speech detection has been performed solely based on text. One state-of-the-art model is BERT. The BERT family (RoBERTa, DistillBERT) is a family of contextualized transformers based on a pre-trained language model which is further fine-tuned for downstream applications such as hate speech classification[5].

For hateful meme classification, the Meta challenge team proposed unimodal training where a ResNext [6] encoder is used for image feature extraction. Apart from this, there has been a multitude of work on extracting information from images, which is potentially useful for hateful meme detection. Image processing systems such as Faster R-CNN or Inception V3 models [7,8] are useful for detecting objects in images.

Most existing multimodal systems adopt either a late-fusion (LF) or an early-fusion (EF) approach to process the two modalities. Late-fusion methods [9] typically utilize unimodal models to process the two signals independently and then combine their features (usually via simple concatenation) before the final classification layer. Models such as MMBT, VisualBERT, and ViLBERT [10,11] use the early-fusion methods and employ more complex approaches to process the two modalities jointly within the model architecture.

In 2020, Facebook conducted the Hateful Meme Challenge. The goal of this challenge was to develop multimodal machine learning models—which combine text and image feature information—to automatically classify memes as hateful or not. Few of these winners of these challenges were substantially able to improve the then SOTA baseline metrics.

The team Detectron that stood 3rd in the competition[12] used an approach that extracted image features using Detectron (an objection detection algorithm) and combined that with a fine-tuned and pre-trained VisualBERT model, They also performed an extensive hyperparameter search by applying Majority Voting Techniques.

The Data scientists that stood [13] second realized that the problem itself was the opportunity of applying the same NLP transformers to images. They pre-extracted the most important content from images first via CNNs such as FasterRCNN (due to hardware limits). The transformer models they implemented then took two different approaches to deal with the extracted images and the text: (A) Two Stream Model: Run them through separate transformers and then combine them at the very end, (B) One Stream Model: Run them in parallel through the same transformer.

Finally, the team that stood first[14] used OCR and an inpaint model to find and remove the text from the image. This improved the quality of both object detection and web entity detection. Using the clean meme image to do bottom-up-attention feature extraction, web entity detection, and human race detection. Those tags gave the transformer models much more diverse information to work with. The models used were extended VL-BERT, UNITER-ITM / VILLA-ITM, vanilla ERNIE-Vil.

2.2 Our innovation

Our approach is different from the above approaches mentioned in the following two ways. Firstly, we have performed PCA on the embeddings of text and images to extract features with most variance and we further experimented with different dimensionalities of these embeddings to understand better how the models are performing. This helped us to produce performant models which are light weight and run faster. This would ensure the ease of integration of our models with other

systems due to their low resource usage. Secondly, we fine-tuned the last 4 layers of RoBERTa pre-trained on twitter's tweets(task: hate) using another data set from UC Berkeley containing hate speech to generate more hate speech "contextualized" embeddings.

3 Dataset and Features

3.1 Dataset

The Hateful Memes Challenge dataset[4] is a dataset and benchmark created by Meta AI to drive and measure progress in multimodal reasoning and understanding. The dataset consists of images and short text which was combined to create a training, validation, and testing set. Meta AI provided around 12000 memes, classified into a train set (8500 memes), a validation set (500 memes), and two test sets of 1000 memes and 2000 memes. Data is provided in JSON files where each line is a dictionary of key-value pairs consisting of data about text and images. The dictionary includes the following keys:

- "id" — This is a unique identifier of the Meme Image.
- "Img" — The folder path of the Meme Image.
- "label" — Label of the Meme Image 0: not-hateful 1: hateful
- "text" — The Text in the Meme.

3.2 Exploratory Data Analysis

Memes contain both text and images. The hateful meme challenge dataset consists of memes containing benign text and benign images which are permuted to convey different meanings and hence different labels.

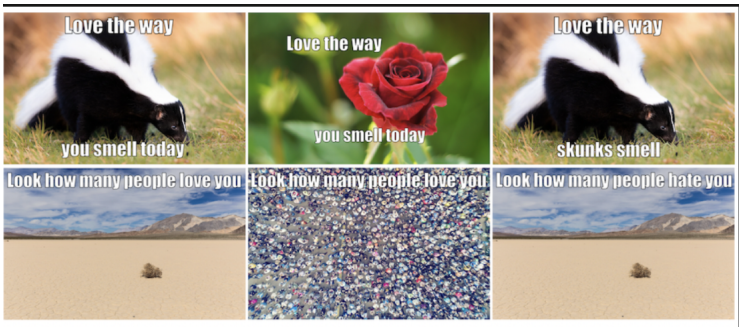


Figure 2: Memes in Dataset

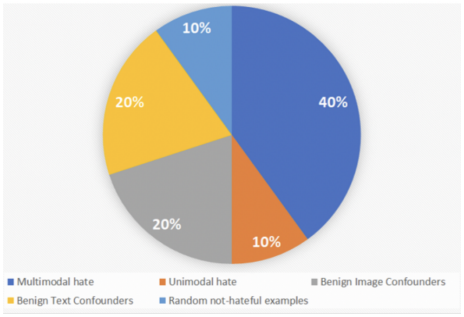
In the image above, both the images have the same text "Love the Way you Smell today". However, when the text is on the skunk, meme becomes hateful whereas when concatenated with the image containing rose, it becomes non-hateful. The same can be done with the text swap on the same image to change the label as shown below.

(a) (b)

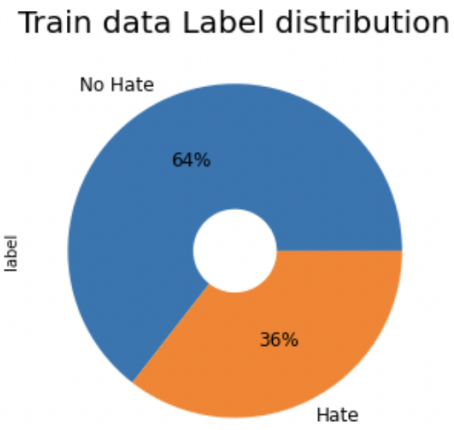
Figure 3: Same image concatenated with different text conveying different meanings.

Analysis of the dataset to understand the skewness of the label distribution reveals that the training set contains 8,500 examples with 36% hateful and 64% non-hateful memes.

The distribution of the five different types of memes in the dataset is as follows: 40% multimodal hate, 10% unimodal hate, 20% benign text, 20% benign image confounders, and the remaining 10% random non-hateful.



(a) Different categories of memes



(b) Training data label distribution

3.3 Additional Data Source

We used an additional data set created by UC Berkeley's D-Lab to fine-tune RoBERTa for the task. This is a publicly released dataset described in Kennedy et al. (2020) [15] consisting of 39,565 comments annotated by 7,912 annotators for 135,556 combined rows. The primary outcome variable being used is "hatespeech" out of the available 10 constituent labels of sentiment, (dis)respect, insult, humiliation, inferior status, violence, dehumanization, genocide, attack/defense, hate speech score benchmark.

3.4 Features and Data Hypothesis

3.4.1 Text based features:

In our dataset, we observe that the texts are of short lengths consisting of 15 words on an average. Our original embeddings were 768 dimensional. Another interesting point is that the memes don't rely on abusive or NSFW (not safe for work) language to be classified as hateful. The meaning and the context (whether it is offensive/hateful) is mostly derived in combination with the image it is used with. For example - one of the inputs are "Obama Voters" as seen below.

Figure 5

3.4.2 Image based features:

Our image based embeddings were 2047 dimensional. There are a few NSFW and pornographical images. Also, our dataset is highly diverse and inclusive:

(a)

(b)

Figure 6: Inclusive dataset

3.4.3 Data Hypothesis:

Since we are using image based data, our inductive biases are rotational invariance (45 degrees rotation in either direction), size invariance, illumination invariance, brightness, contrast, hue invariance, etc. For our text based data, we observed that context matters, and hence we used BERT variants to analyse contextual embeddings.

4 Methodology

4.1 Data Preprocessing

4.1.1 Image Preprocessing

Since our dataset consisted of image based memes with text written in them, we used OCR (Optical Character Recognition) to detect text inside images, and inpainting - the process where missing parts of a photo are filled in to produce a complete image and to subsequently remove the text present in the image. Inpainting is the current SOTA which does include the noise into the images but does not distort their meaning beyond recognition.

4.1.2 Creation of Image and Text Embeddings

In order to perform early fusion, the initial task was to create the correct "contextual" embeddings for text and image. For text embeddings we used SentenceTransformer based on msmarco-roberta-base-v2 which produced a 768 dimensional embedding and for image embeddings we used ResNext50 which produced a 2047 dimensional embedding.

4.2 Baseline Unimodals

4.2.1 RoBERTa [16]

We used Roberta for Sequence Classification which loads classification head or a linear layer on top of the pooled output for binary or multi-class document.The tokenizer we used here is Roberta Tokenizer Fast which is a type of Byte Pair Level tokenizer.

4.2.2 ResNext50 [17]

ResNext (resnext50_32x4d) is a highly modularized network architecture for image classification which replaces the standard residual block with a "split-transform-merge" strategy[18].The transforms used were to normalize and resize the images.

4.3 Multimodals

For our multimodal models, we used an early-fusion based system which combines the different modalities i.e images and text before attempting to classify the content. We attempted to do this using the text embeddings (768 dimensional) generated from RoBERTa and image embeddings (2047 dimensional) generated from ResNext50. This enables us to better detect hateful content even if the image or text are benign individually.

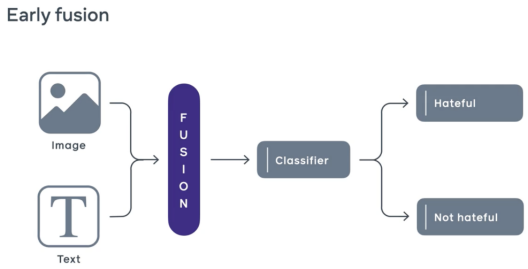


Figure 7: Early Fusion

4.3.1 Preprocessing embeddings using PCA

Since our input was considerably high dimensional, we attempted to use PCA which could help us achieve comparable model performance using less resources and time. Additionaly, PCA helped in reduction of noise in the data, feature selection (to a certain extent), and the ability to produce independent, uncorrelated features of the data. Hence, we experimented with different dimensions to see how the task at hand performs:

1. 2047 dim image embedding + 768 dim image embedding
2. 100 dim image embedding + 100 dim textual embedding
3. 50 dim image embedding + 150 dim textual embedding
4. 150 dim image embedding + 50 dim textual embedding

We tried these different combinations of text and image embeddings for the following models:

4.3.2 Machine Learning Models

Since it is a classification task, it seemed feasible to try the traditional machine learning algorithms and ensembles to analyze how they perform. We picked our hyper-parameters using the grid search technique.

1. Logistic Regression:
   - **Hyper parameters considered:** maximum iterations, regularization penalty, linear classification solver, L1 and L2 regularization ratio.
   - **Optimal:** max_iter=9000, penalty='elasticnet', solver='saga', l1_ratio=0.5
2. XGBoost:
   - **Hyper parameters considered:** the number of estimators i.e the number of decision trees, learning rate, maximum depth of trees, and early stopping.
   - **Optimal:** n_estimators=1000, learning_rate=0.05, objective = "binary:softmax" , num_class = 2, max_depth=6, early_stopping_rounds=50
3. CatBoost:

- **Hyper parameters considered:** the number of iterations, depth of trees, the number of splits for numerical features, tree growing policy which by default is symmetric, etc
- **Optimal:** iterations=100, learning_rate=0.5, loss_function='CrossEntropy', depth=3, l2_leaf_reg = 2, grow_policy= "Depthwise", random_strength = 0.1, od_type = 'IncToDec'

4. LGBM:

   - **Hyper parameters considered:** the number of estimators i.e. the number of decision trees, learning rate, maximum depth of trees, and early stopping.
   - **Optimal:** max_depth= 3, num_leaves=2, num_iterations = 1000, n_estimators = 1000, early_stopping = 50

4.3.3 Deep Learning Models

The deep learning models were fine tuned on number of layers, batch normalization, dropout layers, number of epochs and early stopping.

1. Shallow FFN Model Architecture

```
BNShallowNet(
  (fc1): Linear(in_features=2815, out_features=128, bias=True)
  (fc2): Linear(in_features=128, out_features=2, bias=True)
  (bn): BatchNorm1d(128, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
  (dropout): Dropout(p=0.5, inplace=False)
)
```

Figure 8: Shallow FFN Model Architecture

2. Deep FFN Model Architecture

```
BNDeepNet(
  (fc1): Linear(in_features=2815, out_features=128, bias=True)
  (fc2): Linear(in_features=128, out_features=64, bias=True)
  (fc3): Linear(in_features=64, out_features=32, bias=True)
  (fc4): Linear(in_features=32, out_features=2, bias=True)
  (bn1): BatchNorm1d(128, eps=1e-05, momentum=0.1,
affine=True, track_running_stats=True)
  (bn2): BatchNorm1d(64, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
  (bn3): BatchNorm1d(32, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
  (bn4): BatchNorm1d(32, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
  (dropout): Dropout(p=0.5, inplace=False)
)
```

Figure 9: Deep FFN Model Architecture

4.3.4 Fine-tuning RoBERTa

Finally, RoBERTa was fine-tuned to produce embeddings which are more specific to hate speech detection. UCB DLab's Hate Speech dataset was used for this task. After experimenting with models such as Roberta Base, Roberta for Sequence Classification, and Auto Model For Sequence

Classification, the final model used was Auto Model For Sequence Classification which is pretrained on twitter tweets optimized to detect hate. The tokenizer used for this task was Auto Tokenizer.

The first eight layers of RoBERTa were kept as is and we fine-tuned on the remaining four layers using using UCB's Hate Speech data as our train data and Meta's dataset as our validation set. This model was further used to generate textual embeddings for our early fusion based system.

5 Results

This section highlights the performance of various Machine Learning and Deep Learning Models having varying embedding lengths (with and w/o PCA). In addition to this, we also show the impact of retraining RoBERTA on our model performance.
We fine tuned RoBERTa by using another dataset to generate embeddings which are more specific to our task.
Metric Used: AUC
Loss Function: Cross Entropy Loss

5.1 Unimodals: RoBERTa and ResNext50:

MODEL	Accuracy	F1	AUC	Precision	Recall
ResNext	0.607	0.339	0.540	0.406	0.290
RoBERTa	0.440	0.591	0.560	0.658	0.283

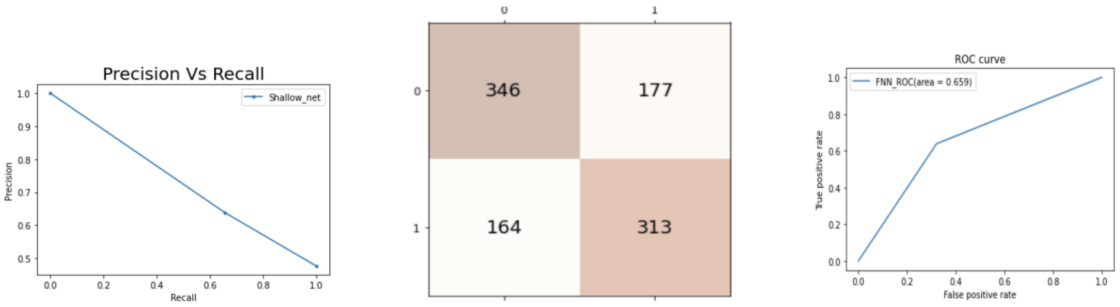
5.2 Mutlimodality without PCA:

Note: TUS stands for Test Unseen and TS stands for Test Seen (Our two different Test DataSets as described above in the dataset section).

Best Model:
DL: Shallow FFN
ML: Logistic Regression
Overall Best Model: Shallow FFN with an AUC of 0.65887

MODEL	Accuracy	F1	AUC	Precision	Recall
Shallow FNN	**TUS:** 0.6565 **TS:** 0.659	**TUS:** 0.5761 **TS:** 0.6473	**TUS:** 0.6427 **TS:** 0.6588	**TUS:** 0.6226 **TS:** 0.6387	**TUS:** 0.5361 **TS:** 0.6561
Deep FNN	**TUS:** 0.669 **TS:** 0.632	**TUS:** 0.4537 **TS:** 0.5269	**TUS:** 0.6431 **TS:** 0.6557	**TUS:**0.3666 **TS:** 0.4183	**TUS:** 0.5952 **TS:** 0.7118
Logistic	**TUS:** 0.631 **TS:** 0.6	**TUS:** 0.4517 **TS:** 0.5110	**TUS:** 0.6341 **TS:**0.6546	**TUS:**0.5100 **TS:** 0.6371	**TUS:** 0.4053 **TS:** 0.4265
CatBoost	**TUS:** 0.616 **TS:** 0.566	**TUS:** 0.4083 **TS:** 0.4506	**TUS:** 0.6127 **TS:**0.6238	**TUS:**0.4835 **TS:** 0.5933	**TUS:** 0.3533 **TS:**0.3632
XGBoost	**TUS:** 0.536 **TS:** 0.618	**TUS:** 0.2723 **TS:** 0.2905	**TUS:** 0.5325 **TS:**0.5292	**TUS:**0.4766 **TS:** 0.5792	**TUS:** 0.1906 **TS:**0.1938
LGBM	**TUS:** 0.625 **TS:** 0.579	**TUS:** 0.3432 **TS:** 0.3977	**TUS:** 0.6228 **TS:**0.6460	**TUS:**0.5 **TS:** 0.6650	**TUS:** 0.2613 **TS:**0.2836

Further Analysis of the best model: Shallow FFN



(a) Precision vs Recall Curve (b) Confusion Matrix (c) ROC curve

Figure 10: Best Model: Shallow FFN (w/o PCA)

5.3 Multimodality with PCA:

5.3.1 100 Dimensional Textual & 100 Dimensional Visual Embeddings:

Best Model:
DL: Deep FFN
ML: CatBoost
Overall Best Model: CatBoost with an AUC 0.59

MODEL	Accuracy	F1	AUC	Precision	Recall
Shallow FNN	**TUS:** 0.429 **TS:** 0.516	**TUS:**0.4352 **TS:** 0.5207	**TUS:** 0.4600 **TS:** 0.5164	**TUS:** 0.5866 **TS:** 0.5367	**TUS:** 0.3459 **TS:** 0.5057
Deep FNN	**TUS:** 0.551 **TS:** 0.533	**TUS:** 0.2815 **TS:** 0.3261	**TUS:** 0.4846 **TS:** 0.5418	**TUS:**0.2346 **TS:** 0.2306	**TUS:** 0.352 **TS:** 0.5566
Logistic	**TUS:** 0.523 **TS:** 0.539	**TUS:** 0.3026 **TS:** 0.3845	**TUS:** 0.4515 **TS:**0.5457	**TUS:**0.3349 **TS:** 0.5559	**TUS:** 0.276 **TS:** 0.2938
CatBoost	**TUS:** 0.5325 **TS:** 0.566	**TUS:** 0.3401 **TS:** 0.4348	**TUS:** 0.4953 **TS:**0.5908	**TUS:**0.3613 **TS:** 0.6007	**TUS:** 0.3213 **TS:**0.3408
XGBoost	**TUS:** 0.566 **TS:** 0.531	**TUS:** 0.2988 **TS:** 0.3193	**TUS:** 0.5021 **TS:**0.5249	**TUS:**0.3790 **TS:** 0.5527	**TUS:** 0.2466 **TS:**0.2244
LGBM	**TUS:** 0.525 **TS:** 0.554	**TUS:** 0.3375 **TS:** 0.4282	**TUS:** 0.4681 **TS:**0.5318	**TUS:**0.3538 **TS:** 0.5758	**TUS:** 0.3226 **TS:**0.3408

5.3.2 50 Dimensional Textual & 150 Dimensional Visual Embeddings:

Best Model:
DL: Shalow FFN
ML: LGBM

Overall Best Model: LGBM with an AUC 0.58

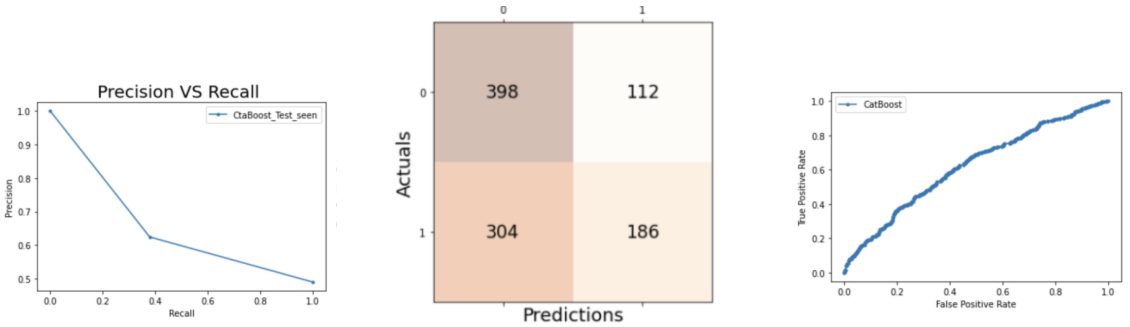
MODEL	Accuracy	F1	AUC	Precision	Recall
Shallow FNN	**TUS:** 0.415 **TS:** 0.550	**TUS:**0.4558 **TS:** 0.5871	**TUS:** 0.4583 **TS:** 0.5541	**TUS:** 0.6533 **TS:** 0.6530	**TUS:** 0.35 **TS:** 0.5333
Deep FNN	**TUS:** 0.5545 **TS:** 0.529	**TUS:** 0.2785 **TS:** 0.3083	**TUS:** 0.4865 **TS:** 0.5369	**TUS:**0.2293 **TS:** 0.2142	**TUS:** 0.3546 **TS:** 0.5497
Logistic	**TUS:** 0.518 **TS:** 0.534	**TUS:** 0.2963 **TS:** 0.3581	**TUS:** 0.4502 **TS:**0.5232	**TUS:**0.3274 **TS:** 0.5508	**TUS:** 0.2706 **TS:**0.2653
CatBoost	**TUS:** 0.544 **TS:** 0.542	**TUS:** 0.3568 **TS:** 0.4113	**TUS:** 0.5017 **TS:**0.5733	**TUS:**0.3787 **TS:** 0.5555	**TUS:** 0.3373 **TS:**0.3265
XGBoost	**TUS:** 0.618 **TS:** 0.529	**TUS:** 0.2723 **TS:** 0.3300	**TUS:** 0.4906 **TS:**0.5232	**TUS:**0.4766 **TS:** 0.5446	**TUS:** 0.1906 **TS:**0.2367
LGBM	**TUS:** 0.5185 **TS:** 0.555	**TUS:** 0.3471 **TS:**0.4121	**TUS:** 0.4742 **TS:**0.5884	**TUS:**0.3531 **TS:** 0.5842	**TUS:** 0.3413 **TS:**0.3183

5.3.3 150 Dimensional Textual & 50 Dimensional Visual Embeddings:

Best Model:
DL: Deep FFN
ML: CatBoost
Overall Best Model: CatBoost with an AUC 0.62

MODEL	Accuracy	F1	AUC	Precision	Recall
Shallow FNN	**TUS:** 0.424 **TS:** 0.532	**TUS:**0.4450 **TS:** 0.5473	**TUS:** 0.4605 **TS:** 0.5331	**TUS:** 0.616 **TS:** 0.5775	**TUS:** 0.3484 **TS:** 0.5202
Deep FNN	**TUS:** 0.5545 **TS:** 0.539	**TUS:** 0.2877 **TS:** 0.3404	**TUS:** 0.4895 **TS:** 0.5501	**TUS:**0.24 **TS:** 0.2428	**TUS:** 0.3592 **TS:** 0.5693
Logistic	**TUS:** 0.5355 **TS:** 0.536	**TUS:** 0.3133 **TS:** 0.3846	**TUS:** 0.4726 **TS:**0.5473	**TUS:**0.3515 **TS:** 0.5492	**TUS:** 0.2826 **TS:**0.2959
CatBoost	**TUS:** 0.533 **TS:** 0.584	**TUS:** 0.3822 **TS:** 0.4720	**TUS:** 0.4988 **TS:**0.6204	**TUS:**0.3792 **TS:** 0.6241	**TUS:** 0.3853 **TS:**0.3795
XGBoost	**TUS:** 0.5775 **TS:** 0.539	**TUS:** 0.3124 **TS:** 0.3328	**TUS:** 0.5132 **TS:**0.5330	**TUS:**0.4008 **TS:** 0.5721	**TUS:** 0.256 **TS:**0.2346
LGBM	**TUS:** 0.5455 **TS:** 0.552	**TUS:** 0.3427 **TS:**0.4105	**TUS:** 0.5020 **TS:**0.5801	**TUS:**0.3744 **TS:** 0.5777	**TUS:** 0.316 **TS:**0.3183

Further Analysis of the best model: CatBoost with 150 Dimensional Text and 50 Dimensional Images:



(a) Precision vs Recall Curve (b) Confusion Matrix (c) ROC curve

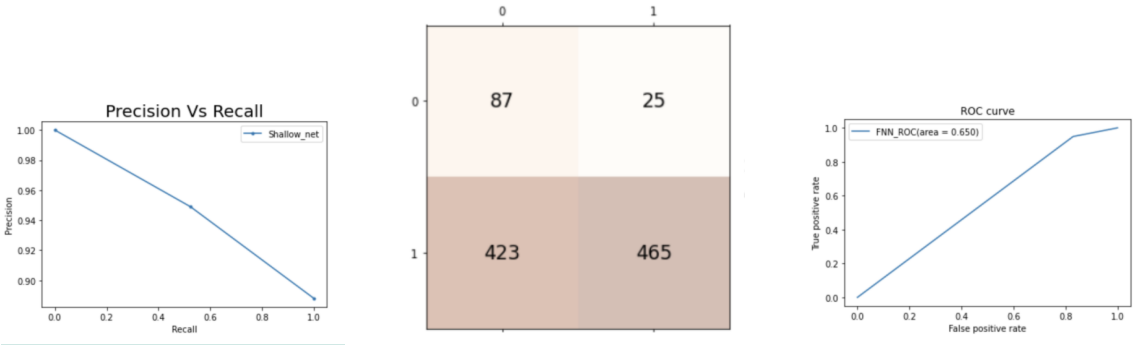
Figure 11: Best Model: LGBM (150 T & 50 I)

5.4 Multimodality with Retrain Roberta (PCA):

Since our learning here was that text was clearly give us more signal as shown in the results obtained by the unimodal and PCA embedding based models, we decided to fine tune RoBERTa in the hope to get customized embeddings.
Best Model with 3:1 text to image embedding ratio:
DL: Shallow FFN
ML: LGBM
Overall Best Model: Shallow FFN with an AUC 0.66

MODEL	Accuracy	F1	AUC	Precision	Recall
Shallow FNN	**TUS:** 0.5374 **TS:** 0.552	**TUS:**0.5426 **TS:**0.6748	**TUS:** 0.3975 **TS:** 0.6602	**TUS:** 0.9533 **TS:** 0.9489	**TUS:** 0.3793 **TS:** 0.5236
Deep FNN	**TUS:** 0.568 **TS:** 0.495	**TUS:** 0.1708 **TS:** 0.1155	**TUS:** 0.4588 **TS:** 0.4550	**TUS:**0.1186 **TS:** 0.0673	**TUS:**0.3047 **TS:** 0.4074
Logistic	**TUS:** 0.5305 **TS:** 0.515	**TUS:** 0.2062 **TS:** 0.2526	**TUS:** 0.4280 **TS:**0.5168	**TUS:**0.2817 **TS:** 0.5157	**TUS:**0.1626 **TS:**0.1673
CatBoost	**TUS:** 0.544 **TS:** 0.544	**TUS:** 0.3467 **TS:** 0.4077	**TUS:** 0.4896 **TS:**0.5901	**TUS:**0.3746 **TS:** 0.5607	**TUS:** 0.3226 **TS:**0.3204
XGBoost	**TUS:** 0.578 **TS:** 0.578	**TUS:** 0.1610 **TS:** 0.1996	**TUS:** 0.578 **TS:**0.5268	**TUS:**0.3164 **TS:** 0.6373	**TUS:** 0.108 **TS:**0.1183
LGBM	**TUS:** 0.495 **TS:** 0.533	**TUS:** 0.3138 **TS:**0.3974	**TUS:** 0.4516 **TS:**0.5477	**TUS:**0.3199 **TS:** 0.5403	**TUS:**0.308 **TS:**0.3142

Further Analysis of the best model: Shallow FFN with 150 Dimensional Text and 50 Dimensional Images:



(a) Precision vs Recall Curve (b) Confusion Matrix (c) ROC curve

Figure 12: Best Model: Shallow FFN (150 T & 50 I)

6 Discussion

Our findings are based on the classification models used over concatenated embeddings of text and "pure images" generated via RoBERTa and ResNext50. The results and discussions also reflect our understanding of the results obtained from the models trained and validated over PCA'd embeddings and those generated via retrained RoBERTa.

6.1 Findings

6.1.1 Metric Used:

We draw analysis on the basis of AUC metric and not simply on accuracy. Since, our dataset , and even in the real world, the data is often skewed i.e there is more non hateful content than hateful content. In addition, there is an implicit skewness in our concatenated embeddings since textual embeddings capture more information than image embeddings i.e it is hard to classify images (without any text) as hateful/non hateful even for human annotators.
Hence, the usage of AUC metric over accuracy makes sure that our models don't overfit.

6.1.2 Capturing the effects of Principal Component Analysis:

- *Without PCA:* The total length of the concatenated embeddings is 2815. We observe here that among the deep learning models, Shallow FFN performed better than Deep FFN. This is because the embeddings are less feature engineered in this case i.e the models are fed in with all the information contained in the embeddings and hence it does not need to capture a complicated relationship for the classification task at hand. Hence, the Shallow FFN is sufficient for the task and probably Deep FFN in such a case is overfitting and hence not giving good results.
  Also, among the machine Learning models, we observe that Logistic Regression is performing better than Ensemble methods. Since, ensembles use weak classifiers, such as decision trees which perform splitting of nodes based on features, they fail to classify the content correctly

given the high dimensional embeddings i.e. high number of features.
Overall, the Shallow FFN performs the best with an **AUC of 0.65**

- *With PCA:* We observed that a 200 dimensional concatenated vector was capturing most of the information (i.e the most variance) of our original 2815 dimensional vector. We experimented with the weightage given to text and image embeddings i.e the individual lengths of the image and text embeddings as follows:
  - *100 dimensional image and 100 dimensional textual embeddings:*
    Best Model: CatBoost with an **AUC of 0.59**
    DL : Deep FFN
    ML: CatBoost
  - *150 dimensional images and 50 dimensional textual embeddings:*
    Best Model: CatBoost with an **AUC of 0.58**
    DL : Deep FFN
    ML: CatBoost
  - *50 dimensional images and 150 dimensional textual embeddings:*
    Best Model: LGBM with an **AUC of 0.62**
    DL : Shalow FFN
    ML: LGBM
- PCA helped us reduce noise, removed correlation between features and helped with feature selection, however, it led to a slight deterioration in the performance because of loss of information in the input embedding.
- The trend observed here is that the Deep FFN is performing better. This is because the features are now engineered (i.e features are now projected in a lower dimensional space) and hence the model needs to capture more complicated relationships to correctly classify the memes which the deep FFN is able to do.
- Also, since ensembles methods here are being fed less number of features, they are able to select features more efficiently and hence are performing better.
- Out of all the combinations of lengths, we observe that models are performing best with *150 dimensional textual embeddings + 50 dimensional image embeddings*. This follows from the fact that the textual embeddings indeed give us more signal.
- Another interesting observation is that ensembles (with PCA) are actually defeating Deep Learning models. This tells us that clever feature engineering when combined with machine learning gives better results especially for classification tasks.

6.1.3 Retrained RoBERTa:

Since, textual embeddings were giving more information, we decided to fine tune RoBERTa to get better embeddings. We tested the model using 150 dimensional textual embeddings + 50 dimensional image embeddings since these worked best in the previous models. Shallow FFN model performs the best in these setting and gives **AUC of 0.66**. Here, Shallow FFN is working better than Deep FFN since the embeddings were already fine tuned in accordance to the task at hand i.e hate speech.

6.1.4 Test Seen vs Test Unseen:

On observing the trend of AUC over Test Seen and Test Unseen data sets, it was noted that Test Seen reported a higher AUC across all the models. The reason being Test Seen set has a higher overlap of memes with the train dataset as compared to the Train Unseen set. This implies that model is able to learn and predict the memes it has already seen correctly.

6.1.5 Comparison with our Unimodal baselines:

Unimodels reported a low AUC of 0.54 (ResNext50) and 0.56 (RoBERTa). This was expected as the dataset used for this project was curated in a way that the unimodels were bound to fail at classifying memes accurately. Our final model performed better when more weightage was given to the text embeddings. This result matched with the intuition that we got from unimodels i.e. the text is providing much more signal than images.

6.1.6 Social Impact

The debate around the regulation of hate speech is still ongoing [19]. It is still not clear whether the best response to it is through legal measures, or other methods (such as counter-speech and education). Regardless of the means of countering it, the evident harm of hate speech makes its detection crucial. Both the volume of content generated online (particularly on social media) and the psychological burden of manual moderation highlight the need for the automatic detection of offensive and hateful content.
The uncontrolled spread of hate has the potential to gravely damage our society, and severely harm marginalized people or groups[20]. A major arena for spreading hate speech online is social media. This significantly contributes to the difficulty of automatic detection, as social media posts need models to take into account multimodality of media.
Furthermore, such a model is not restricted to detect hate speech in only memes. Its usability can be extended to any domain dependent on multimodal media such as posters and hoardings etc.

6.1.7 Why the results matter and the difference made:

We have attained success at re-training a pre-tuned RoBERTa (on "hate" task) which helped in attaining better performance on the task by generating embeddings which are more tailored to the task at hand. This can be extended to many "tasks" defined for AutoModel such as sentiment analysis and all such models can be stacked to get a weighed embedding for the same text. This can help us get the embeddings which define the relative proportions of tasks/emotions in them. Also, our findings prove that PCA does not provide much loss of the AUC. This shows that there is a possibility of achieving good classification with a low resource intensive model. Hence, while trying to aim for models which are robust and performant we can consider doing PCA on the embeddings for the best model attained.

6.2 Limitations and Ethical Considerations

6.2.1 Limitations

The difficulty of automatic detection, as social media posts, include paralinguistic signals (e.g. emoticons, and hashtags), and their linguistic content contains plenty of poorly written text. An-

other difficulty is presented by the context-dependent nature of the task. This can be societal and cultural context. Hence, the lack of consensus on what constitutes hate speech makes the task difficult even for humans.

6.2.2 Ethical Considerations

The dataset on which the model is trained and tested is very well proctored, unbiased, and well balanced to contain a good mix of neutral and hateful images with neutral and hateful text which may or may not be sarcastic/hateful. The images are also inclusive and contain people of color, whites, children, animals and inanimate objects. The images are also inclusive of all the genders - male, female, queer people. Following are some examples.

(a) People of color in neutral meme.

(b) People of color in neutral meme.

(c) White People in hateful meme.

(d) White child in hateful meme.

Figure 13: Inclusive dataset

However, the model could be misused to propagate racism. This can stem from annotators' insensitivity to differences in dialect which can lead to racial bias in automatic hate speech detection models, potentially amplifying harm against minority populations

6.3 Future Research Directions

Detection of hate in text and images is an open field of research. While we can gain signals from each individually, the combined intent of both can be completely opposite of what is perceived from each individually. With model accuracy being a concern, such a model should also be robust to work on low resources and also, be omniscient of the cultural and societal context. As part of the future scope, one can experiment with Vilbert and VisualBERT as the multimodal models given more compute power and resources. The above mentioned limitations can be combated using Transfer Learning Models. Robustness of a model can also be defined as its capability to not only detect one particular emotion but to provide the relative proportion of different emotions/intents being conveyed such as happiness, sarcasm, threat etc. The model should also evolve to be able to parse emoticons pasted on top on images and to understand multiple images pasted together to form final image/context. These robust features would make the model viable in various domains such as advertisements, linguistic studies etc. In the world of advertisement it would help by providing a cultural inference check of the implied meaning being conveyed. In the linguistic domain, it will help experiment with overlap between languages to be able to train a model which is able to detect emotions in text from various languages.

7 Conclusions

- Unimodal priors are not enough and there is a dire need of models that understand multimodality.
- A faster, portable and light weight model can be generated without much loss on performance by projecting the embeddings to lower dimensions using Principal Component Analysis.
- It can be observed that text provides more signals/information as compared to the images. This can be seen in the results obtained from our unimodels as well as PCA embeddings with ratio of text:images of 3:1.
- Further, it can be observed that Shallow FFNs are better than Deeper FFNs when the relationship to be captured is less complicated i.e. when these models are fed with the non-PCA embeddings or the embeddings obtained from the Retrained RoBERTa.
- It is noted that deep networks perform good with lower dimensional embeddings i.e. post PCA, as the information is now compressed and in the form of linear combination of the features. Now, the mdoels are needed to learn a more complicated function to correctly classify the memes and hence the Deep FFN are able to outperform other models.
- Another interesting learning is our ensembles actually perform better than deep learning models with PCA based embeddings. This shows us that feature engineering combined with machine learning models when fine-tuned properly can actually outperform deep learning models.
- Finally, RoBERTa when fine tuned (using another hate speech based data set) is able to generate embeddings which are better specifically tailored to our task and thus the models are able to perform better at their classification.

8 References

1. Savvas Zannettou, Tristan Caulfield, et al. On the Origins of Memes by Means of Fringe Web Communities. arXiv:1805.12512 [cs.SI].
2. Stanislaw Antol, Aishwarya Agrawal, et al. VQA: Visual question answering. In Proceedings of the IEEE international conference on computer vision.
3. Desmond Elliott, Adversarial evaluation of multimodal machine translation. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing.
4. Douwe Kiela et al, The Hateful Memes Challenge: Detecting Hate Speech in Multimodal Memes,2019, arXiv:2005.04790.
5. Devlin et al., BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, NAACL 2019.
6. Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, Deep Residual Learning for Image Recognition (10 Dec 2015).

7. Ren et al.,Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, 2016.

8. Szegedy et al.,Rethinking the Inception Architecture for Computer Vision, 2015.

9. Douwe Kiela, Hamed Firooz, et al., The hateful memes challenge: Detecting hate speech in multimodal memes. 2020, arXiv:2005.04790.

10. Douwe Kiela, Suvrat Bhooshan, Hamed Firooz, et al, Supervised multimodal bi transformers for classifying images and text., 2019, arXiv:1909.02950.

11. Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee, Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks, Advances in Neural Information Processing Systems, 2019.

12. Riza Velioglu and Jewgeni Rose, Detecting Hate Speech in Memes Using Multimodal Deep Learning Approaches: Prize-winning solution to Hateful Memes Challenge, 23 Dec 2020.

13. Niklas Muennighoff, Vilio: State-of-the-art Visio-Linguistic Models applied to Hateful Memes, 14 Dec 2020.

14. Ron Zhu, Enhance Multimodal Transformer With External Label And In-Domain Pre train: Hateful Meme Challenge Winning Solution, 2021.

15. Chris J. Kennedy, Geoff Bacon, Alexander Sahn, Claudia von Vacano, Constructing interval variables via faceted Rasch measurement and multitask deep learning: a hate speech application, 22 Sep 2021.

16. Yinhan Liu, Myle Ott, RoBERTa: A Robustly Optimized BERT Pretraining Approach, 26 Jul 2019.

17. Saining Xie, Ross Girshick, Aggregated Residual Transformations for Deep Neural Networks, 2017.

18. Christian Szegedy, Wei Liu et al, Going Deeper with Convolutions, 2014.

19. Alkiviadou N. The legal regulation of hate speech: the international and European frameworks. Politička misao. https://doi.org/10.20901/pm.55.4.08.

20. Kovács, G., Alonso, P. Saini, R. Challenges of Hate Speech Detection in Social Media. SN COMPUT. SCI. 2, 95 (2021).
https://doi.org/10.1007/s42979-021-00457-3.