**STATS 402 - Interdisciplinary Data Analysis**

**<Hateful Meme Detection>**

**<Yijia Xue; Ruitian Wu>**

**<**[**yx179@duke.edu**](mailto:yx179@duke.edu)**;rw219@duke.edu>**

##### Abstract

Currently, hateful memes have become one of the most severe issues. On the one hand, hateful memes include information promoting discrimination and prejudice, which could cause society instability. On the other hand, since memes combine texts with images and could contain implicit information, it’s hard for traditional methods to detect it easily. Therefore, we are motivated to develop a system which could detect hateful memes instantly. To achieve our goal, we first proposed text-based unimodal models and an image-based unimodal model. Also, to improve upon the original ones, we implemented a model that first detects tags regarding objects, human races, genders, and emotions from the image. Then we appended the new tags into the original texts. Lastly, we have a multimodal model as our final proposed model. We compared the results between baseline ones and our own proposed one in terms of accuracy, precision, and recall. We figured out that generally accuracy, precision and recall could increase by 4.2%, 2.4% and 12.1% compared with unbiased baseline models. This concludes that our proposed feature expansion method improves classification. However, limitations including imprecise tags and poor image feature quality still exist and therefore we proposed some future work to deal with them.

##### 1. Introduction

##### 1.1 Background Information & Motivation

Hate speeches are regrettably omnipresent today. Hate speech exists not only in offline communication but also in online communication. Because of the anonymous nature of communication in virtual worlds, people feel less responsible for what they say and express more hurtful things. The spread of hate speech will promote discrimination and prejudice and increase the frequency of violent and terrorist incidents in real life. For example, studies have shown that before hate-related terrorist events, terrorists will have posted hate speech on social media for quite a long time[1]. Because hate speech has a negative impression of social stability, there is significant motivation to explore hate speech detection.

In today’s social media, however, text is not the only format for voicing ideas publicly. Nowadays, one most popular format that is employed by people, especially the younger generation, is the meme, which poses an interesting multimodal fusion of both image and speech. Originally, memes are meant to express some ideas in a humorous or implicit way. The meaning expressed by memes could be so concise or even excellent that people love to use it anywhere. However, it could also be used to show irony or even hatred towards others, which might make it harmful. Nowadays, when people are trying to express something hateful, they would turn to hate memes more instead of hate speech to be less likely to be discovered by administrators of social media[2].

Unlike simple hate speech which could be detected with some certain Natural Language Processing methodologies, hate memes are hard to be captured. For example, both the image and the speech might be innocuous when we view them individually. However, if we combine them together, they could mean something ironic and thus harmful. For example, if we gave a speech “Look how many people love you” and an image of a dessert, neither of them seems to be harmful. However, if we combine them together, this could be sarcasm indicating that there are no people loving you at all, which is quite offensive [3]. The example could be shown in Figure 1. Sometimes, it’s even hard for humans to extract features from those memes and judge whether they are harmful or not. Therefore, it’s necessary to develop a system that could extract key information from the memes and help people to make judgments.



Figure.1: Example of Hateful Meme

Different people have different opinions on what memes should be seen as hateful. In our project, we followed the definition given by Facebook Hate Speech Transparency Center [4]:

“Hateful meme” refers to

1. The meme contains direct or indirect attack on people based on characteristics, including disability, disease, sexual orientation, gender identity, sex, caste, religion, immigration status, race, nationality, and ethnicity.
2. The meme contains attacks such as dehumanizing speech or violence, calls for exclusion or segregation and statements of inferiority.
3. The meme includes mocking hate crime.”

Once the hateful meme system is developed, it could be quickly applied to solve social problems: Hatred towards others on social media. Once the memes are detected, administrators of social media like Facebook or Twitter could quickly delete the contents and give penalties to those who post them to prevent a large scale of discrimination or hatred online. What’s more, the experience gained from the task could be further applied to some other real-world multimodal machine learning tasks, which could drive the progress of the new discipline.

**1.2 Characteristics for the method**

We divided our project into constructing three types of models: text-based model, image-based model, and multimodal model. The main architecture of the project is shown in Figure 2.

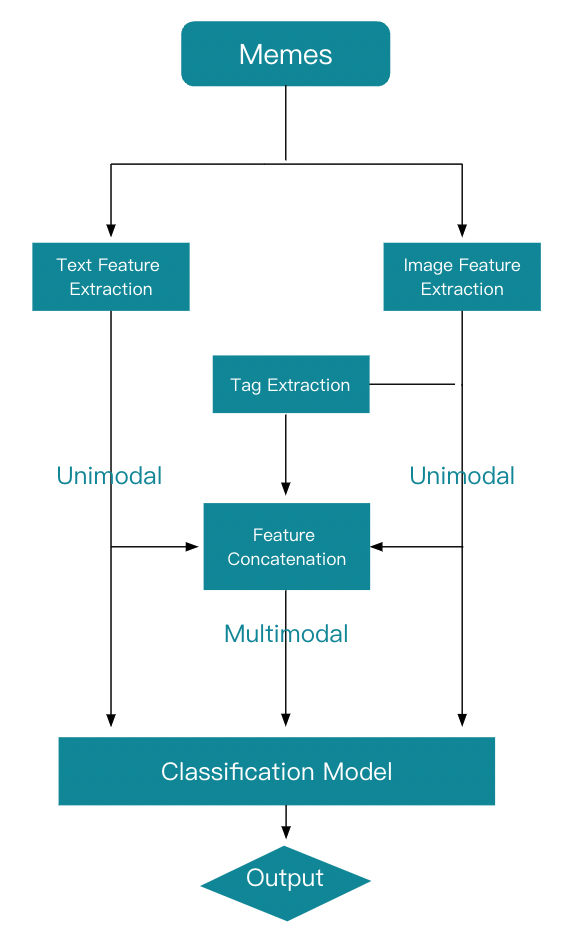


Figure.2: Flow chart of the project

According to the flow chart, our project is mainly divided into 5 steps: ​​

I. Extract text features and implement text-based model

II. Extract image features and implement image-based model

III. Extract tag with hate-related information

IV. Concatenate three channel’s input

V. Implement the final classification model

Then, the goal of the proposed method is to finish the task of each step.

For step I, we implemented

1. TF-IDF based model [5], which extracts features measuring importance of each word. Then some naive Machine Learning based classifiers are implemented based on the extracted text features.
2. Text-CNN based model [6], which captures the local features for the text and usually performs well for short texts.
3. BERT based model [7], which tokenizes and encodes the whole sentence and uses the transformer to do the classification. It performs well on most of the natural language processing tasks.

For step II, we implemented a Resnet-based model [8], it uses Resnet 50 as the backbone model and does the classification by the SoftMax layer. Residual network has been successfully implemented in various computer vision tasks [9].

For step III, we implemented the Yolov5 model [10] to do the object detection and thus can extract the object tag. Haar Cascade Classifier is used to extract the facial bounding box from the initial image for further tag extraction [11]. Pre-Trained face classification model ‘Demo Classi’ is used to detect the face type and emotion type of the given face, which performs well on the pre-trained dataset [12-14].

We implement the Visual-Bert based classifier [15] to accomplish the task of step IV and step V. Compared with the unimodal model, the final classifier concatenates three channels’ input and thus pour more useful information to the model.

**1.3 The organization of the subsequent part of the article**

In the subsequent part of the article, we will first summarize the existing method for this problem and explain the difference between our approach with the existing methods. Then, we will introduce our proposed method according to the sequence of the project flow. Furthermore, we will demonstrate the experimental results of our proposed method and analyze them. Finally, we will summarize the project and describe the plan work.

# 2. Related work/literature review

Currently, regarding hateful speech only, there have been quite a lot of studies that have some satisfactory results. For the models with good performance, they usually combine some advanced feature extraction techniques with traditional machine learning algorithms. For example, C´ ecillon et.al [16] proposed the method of merging content and graph-based features and implementing the hybrid fusion SVM. Their classification result achieved 93.26% F measurement.

However, hateful meme detection is far much more complicated than hateful speech detection in that many memes could become hateful only if both texts and images are being taken into consideration. Therefore, although hateful speech detection techniques achieve some success, it could not be directly transferred to hateful meme detection. Douwe Kiela et. al [17] proposed the state-of-art text detection algorithm BERT [7] on texts in memes to see if detecting texts only could distinguish hateful memes and non-hateful ones. The maximum accuracy that they could achieve is only about 58%. Also, if they classify on images only using the ResNet [8] or Fast-CNN [18] framework, the accuracy could become even worse, about 52.67% at most. For those unimodal detection trials, the biggest advantage is that there are already plenty of studies working on accurate feature extraction of images or texts and it’s easy to directly apply some machine learning algorithms on them. However, the problem is that unimodal models could only capture limited parts of the features which is not enough to make object decisions on meme classification.

Therefore, to collect enough information from the memes, multimodal models that combine both text and image features are proposed. However, surprisingly there has been little work regarding the multimodal hateful meme detection. In the article by Douwe Kiela et. al [17], they proposed a plain multimodal methodology which directly concatenates the result of BERT and ResNet and takes the mean score. Although the model is easy to implement and could reflect the combination of two modalities, its accuracy is still not satisfactory, which is 59.39%. Then, methods of concatenating the two modalities on the feature level instead of a decision level were proposed. For example, Apeksha Aggarwal et al. [19] proposed methods of extracting image features from ResNet 152 and text features from Fast Text and concatenating features to build a Multilayer Perceptron. The model could also only have an accuracy of 59%. Then, more sophisticated models that could decode images and texts simultaneously were employed, including Visual-BERT [15] and VL-BERT [20]. In the article from Zhang et al. [21], they claimed that the two models could separately have an accuracy of 64.3% and 65.3%, which has obvious improvements compared to previous ones.

However, we assume that the models could not give an incisive understanding to the features especially for the image ones. In those multimodal, image features are just extracted as a whole, with no additional understanding of the images. However, from our perspectives, some features that could describe real-world scenarios should be extracted. Instead of understanding “what’s the image like”, it’s more useful for us to understand “what story is the image telling”.

Therefore, compared to previous works, we proposed a new idea which aims to add a new channel to the original two channels: tags. Before combining image and text features, we add an additional step to extract some tag information describing present objects, human races, genders, and emotions. The new tags will be then fused into texts to construct the multimodal model. This methodology could be quite reliable since object detection and human face classification models have been quite intelligent nowadays. For example, for one of the most famous Object Detection YOLO [10] which is both incredibly quick and accurate. Also, face classification models like FaceNet [22] could make clear recognition and clustering of human faces as well. Therefore, we could believe that our newly obtained tags could be quite useful and reliable.

To conclude, the biggest difference of our proposed method lies in feature expansion rather than improvements of the model itself. We believe that the full utility of data is a non-negligible factor in the project. The detailed methods that we implement will be introduced in detail in the next section.

# 3. The proposed method

In this section, we will introduce all proposed methods for the task as we mentioned in the introduction section.

**3.1 Text-based unimodal model**

The goal of the Text-based unimodal model is to accomplish a sentence classification task: whether sentences in a meme include hateful information. We have constructed and implemented 3 different models: machining learning-based classifier model with TF-IDF text feature extraction method, BERT-based classifier model, and Text-CNN based classifier model. The implementation details will be omitted in this paper due to the page limit. We will only introduce the mechanism of the model.

**3.1.1 TF-IDF based model**

Firstly, we used the TF-IDF technique to deal with our text data. For traditional machine learning algorithms, it could only deal with categorical or numerical variables. Texts, mainly made up of strings, could not be directly added to traditional machine learning models. Therefore, we should transform the text files into data types that could be easily processed. That’s the basic rationale for the TF-IDF algorithm [5].

TF-IDF is the abbreviated form for “Term Frequency-Inverse Document Frequency”, which is used to measure the frequency of each word in a document and further in a collection. It is recognized as numerical statistics to measure the importance of each word [5]. In TF-IDF, the importance of a word increases in proportion to its occurrence in a document and decreases with its occurrence in the whole collection. The formula could be constructed as below:

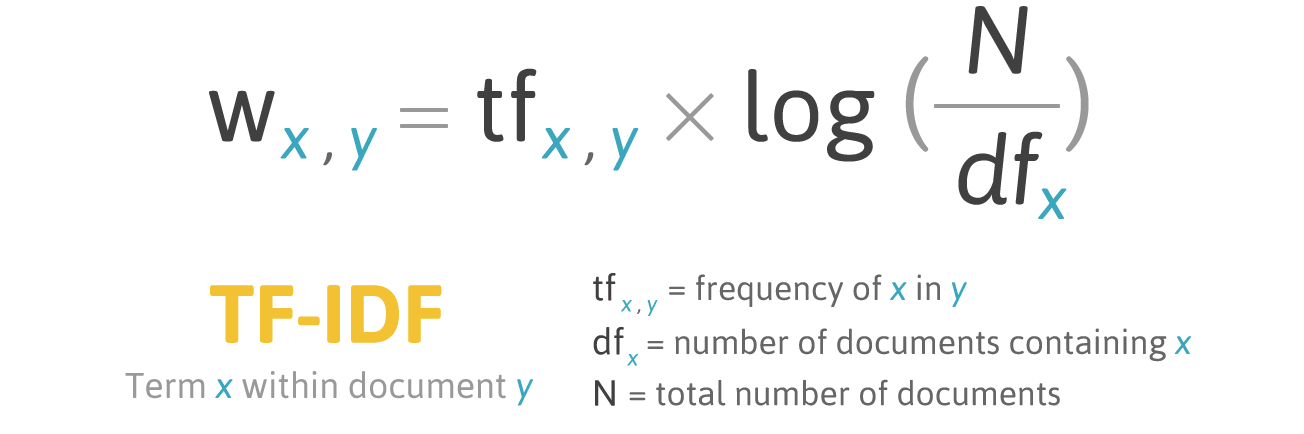


Figure.3: Formula for TF-IDF

Retrieved from <https://ted-mei.medium.com/demystify-tf-idf-in>

-indexing-and-ranking-5c3ae88c3fa0

After obtaining the tf-idf matrix for model building, we directly applied some Machine Learning algorithms, including Logistic Regression, Support Vector Machine, Random Forest, Naive Bayes and K Nearest Neighbors. These algorithms are selected since they are on the one hand easy to implement and on the other hand have a quite stable performance on classification tasks.

**3.1.2 BERT-based model**

Bidirectional Encoder Representations from Transformer (BERT) model is a natural language processing framework [7]. The goal of the BERT model is to use large-scale unlabeled corpus training to obtain text representation containing rich semantic information [7]. BERT is a pre-trained model which means that we can directly use BERT's feature to represent the word embedding feature in our sentence classification task. To solve the sentence classification task using the BERT model, the Bert model should be adjusted as follows: add [CLS] symbol to the input sequence to enable BERT to accept statement-level input; [CLS] is used to represent a semantic representation of the text.

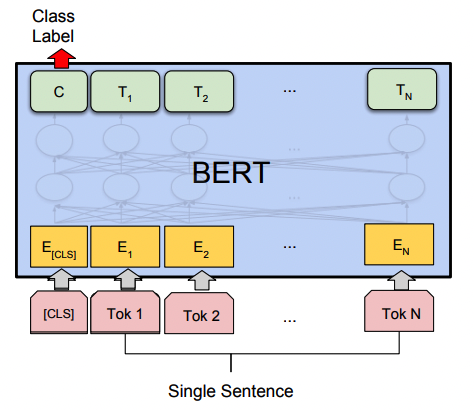


Figure.4: Illustration of Fine-tuning BERT on sentence classification task

Retrieved from https://arxiv.org/pdf/1810.04805.pdf

The BERT-model will first tokenize the sentence and then encode the inputting words. An illustration of sentence tokenizing is given below.

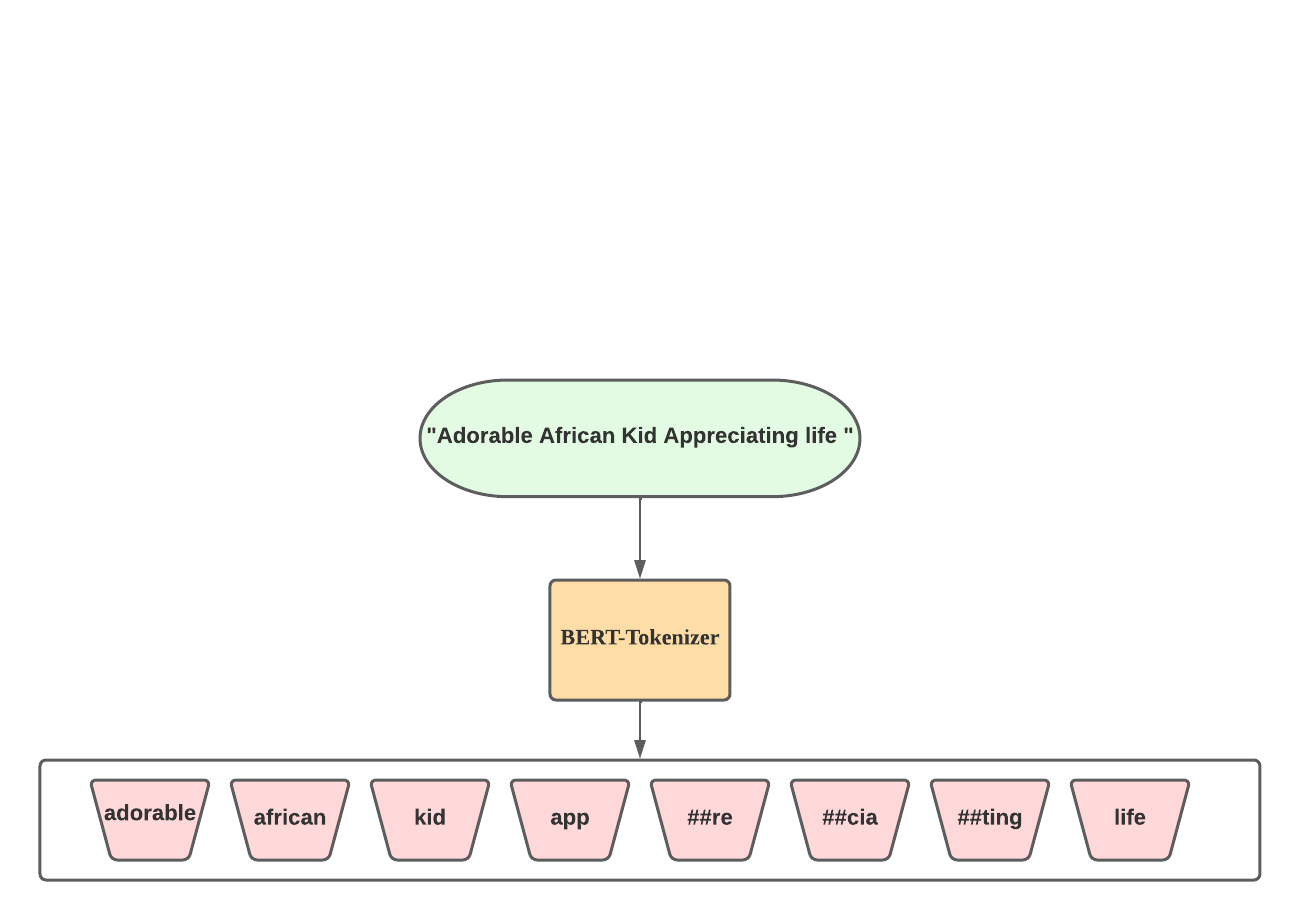


Figure.5: Illustration of sentence tokenizing

Data encoding process gives token\_id to the input tokenized words respectively. For example, the token\_id for “adorable african kid app ##re ##cia ##ting life” is [23677, 3060, 4845, 10439, 2890, 7405, 3436, 2166].

**3.1.3 Text-CNN based model**

The main idea of Convolutional Neural Network (CNN) is to capture local features by using the convolution kernel [6]. The text feature is different from the image feature, the local text feature captured by the kernel is the receptive field made up of several words [6]. Semantic information at different levels can be obtained by CNN using different combinations of text feature filters. For the Text-CNN used by this project, we construct the network by 3 basic blocks: word embedding block, convolution block, and output block. An illustration of the Text-CNN configuration is shown below.

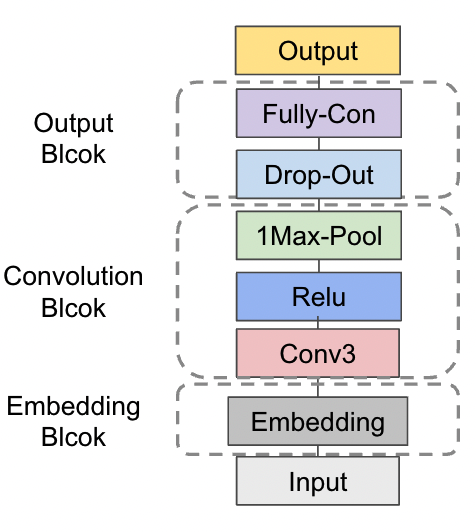


Figure.6: Text-CNN configuration in the project

**3.2 Image-based unimodal model**

Among all feasible image-based models, the ResNet structure built upon Convolutional Neural Network tends to outperform among all other algorithms. Usually from experience, for a neural network, the feature it could extract would be more accurate and more complicated if more layers are added to the architecture. However, when numbers of layers increase significantly, a degradation problem would bump out which would cause deeper networks to have even lower accuracy than the shallow one [8]. For ResNet, it employs gates that could train the residual of inputs and outputs to increase the number of layers in a model, which could be shown in the following figure [8].

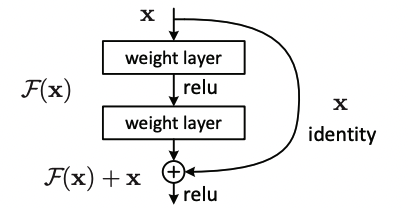


Figure.7: Residual Learning Unit

Retrieved from <https://arxiv.org/pdf/1512.03385.pdf>

**3.3 Multimodal model**

The structure of the multimodal model is shown below

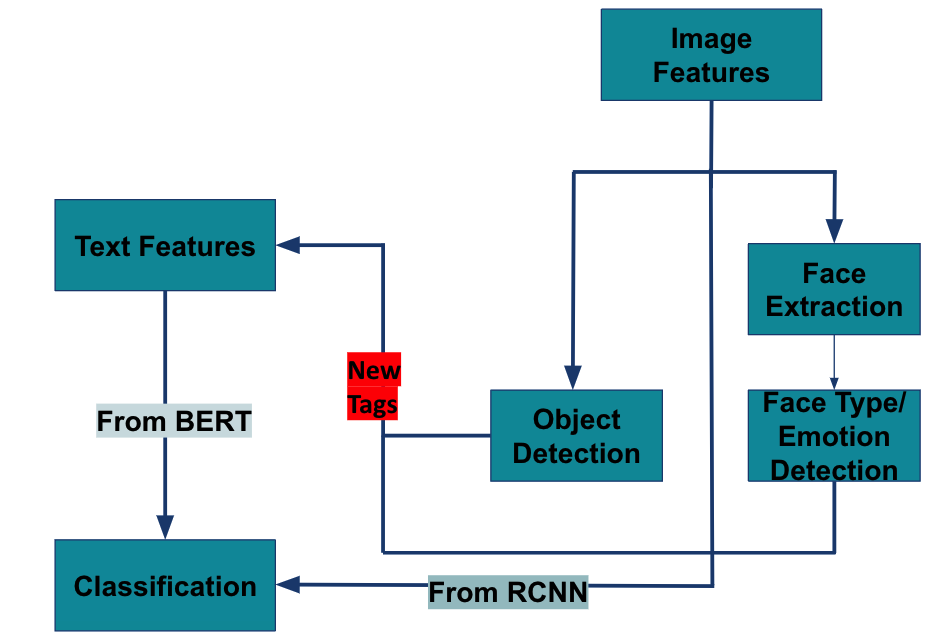


Figure.8: Structure of the Multimodal model

There is one indispensable step in the construction of the final multimodal model: to derive the hate-related tags from the image. Since one type of the hateful meme refers to direct or indirect attack on people based on characteristics. Then we want to extract human’s characteristics such as gender, race, and emotion from the meme.

The task of extracting information from human is divided into 2 steps:

1. Labeling images that contain human faces (3.1.1)
2. Extract characteristics information from the labeling images (3.3.2)

Moreover, the definition of hateful meme also mentions dehumanizing information. We then implemented an object detection model to detect what objects exist in the meme. The purpose of this procedure is to find the internal relation between the meme scene and the dehumanizing information. We will introduce the methods in detail in 3.3.3.

**3.3.1 Face Recognition Labeling with Haar Cascaded Classifier**

To do the face labeling, we must first locate the Region of Interest for each face on a particular image and crop them from that image. To do so, we employed a Face Detection algorithm called Haar Cascaded Classifier [11]. For the Haar Cascaded Classifier, the core basis is to find Haar-like features in each image (See Figure.9) [11].

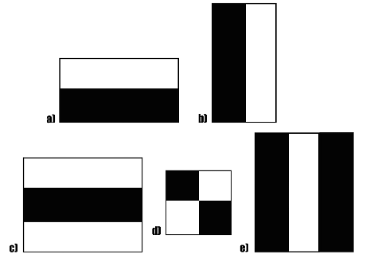


Figure.9: Haar-like Features

For the Haar-like features, it tries to make comparisons of values between adjacent rectangular areas, which could further compare the lightness of different regions. For example, a rectangular with its upper region black and lower region white could indicate that usually eyes are blacker than noses and therefore the region could be seen as one feature describing faces.

It is true that Haar-like features could be quite efficient while extracting useful features of faces from an image. However, the issue is how to determine the best features from a huge pool of potential features. It’s too expensive to include all features into our classifiers. Therefore, we are using a cascaded classifier technique in which we are building a decision tree to gradually reject all false sub-windows with weak classifiers built on single features. The structure could be seen below in Figure. 10. In the cascaded classifier, we input all potential sub-windows into the model and in each stage, the classifier will decide whether to accept that sub-window or not. If accepted, the sub-region will be put into the next classifier until the final Region of Interest is determined by the system [11].

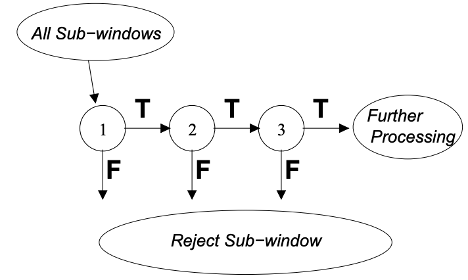


Figure.10: Structure of Cascaded Classifier

With a pre-trained cascaded classifier, we trained a model that could find the regions that are most likely to be human faces. Examples of extracted faces from images could be seen below.



Figure.11: Example of Extracted Faces

However, another issue is that within each image, there might be multiple redundant faces extracted. For example, for the below image, the faces of onlookers will be counted by our model as well, which might dilute our real important information, like the face in red rectangle. To extract the potential most important features from faces, we use the face which takes the largest proportion of area in the image to represent the image.

By running the algorithm, we extracted the most important faces from all 12141 images and finally got 8306 faces as our output.

**3.3.2 Face Type and Emotion Detection**

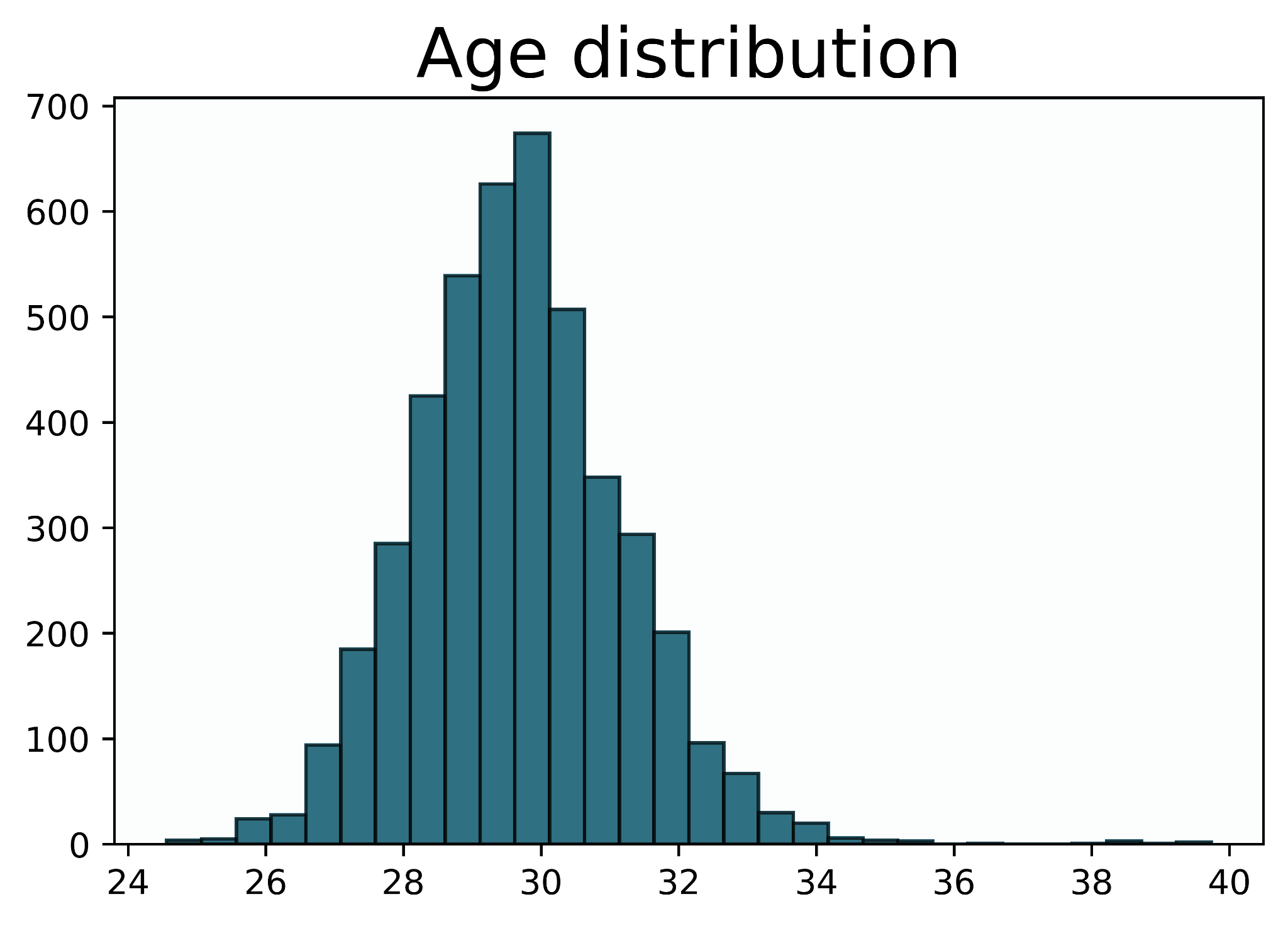
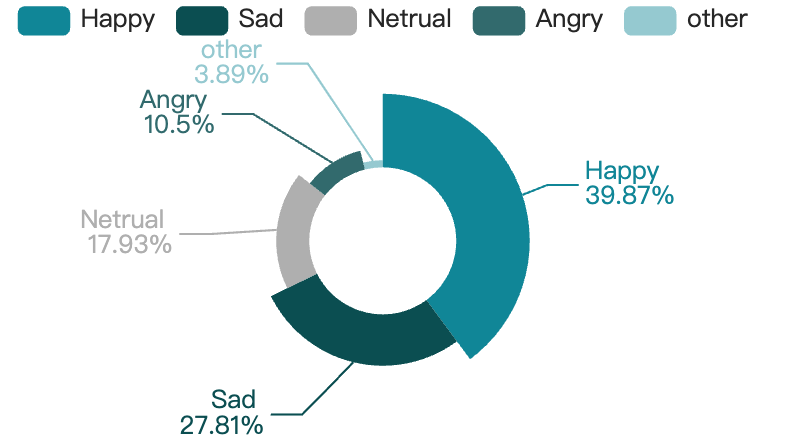
For the face type detection process, we aim to detect the emotion, gender, race, and age from facial images. Since we do not have these demographic labels for the memes, we adopted two pre-trained models and implemented them without fine-tuning. The face type detection model has been pre trained on the UTK *face dataset* [14] and the emotion detection model has been pre trained on the *Fer2013 dataset* [13]. We choose the model with the backbone of Resnet since it performs better than the other model on the evaluation dataset. An illustration of implementation architecture is shown as follows.

图示

描述已自动生成

Figure.12: An illustration of model input and output

Among 8306 memes with face, the distributions of emotion, age, gender, and age are shown below.



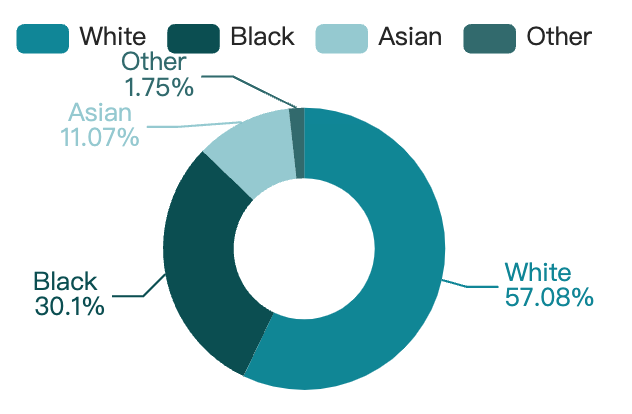
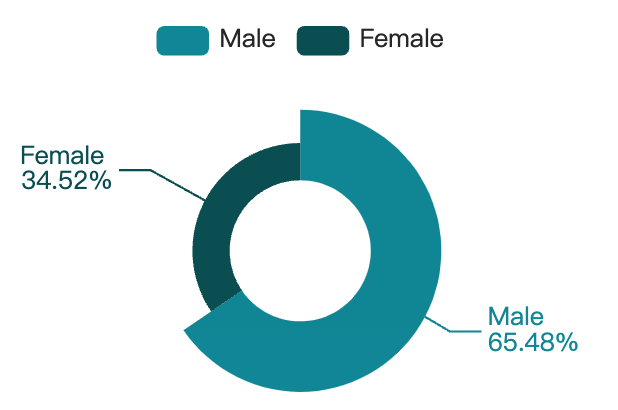


Figure.13: Data distribution of different tags with the sequence (Emotion, Age, Gender, and Race)

We find that the detected age of people in the meme are mainly in the range [25,40] which makes them less distinguishable from each other. This result implies that the detected age tag is useless compared with other tags. So, the age tag is wiped out from the tag set.

**3.3.3 Object detection based on YOLOv5**

There are plenty of Object Detection frameworks in the discipline and in our project, we turned to the model YOLOv5 which is faster and has better precision compared to other frameworks. The basic structure of the YOLOv5 model could be visualized as below [10]:



Figure.14: Structure of YOLOv5

Retrieved from: <https://github.com/ultralytics/yolov5/issues/280>

With a pretrained YOLOv5 structure built upon COCO dataset

[23], we extract tags of all detected objects in all 12151 images.

In our results, we obtained in total 149 objects which seems to be a huge source of features for our model. The distribution of results could be seen below.

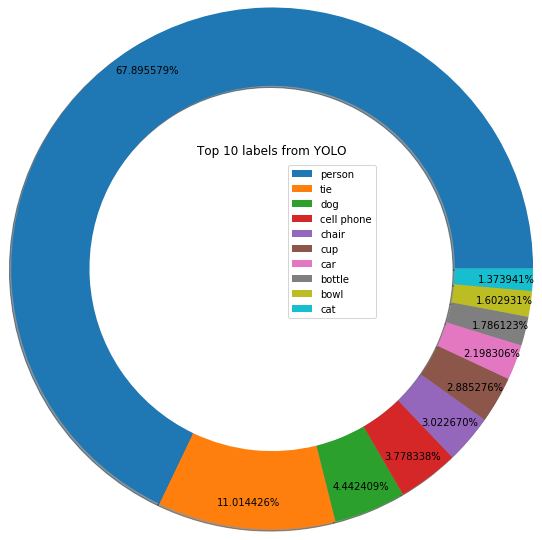


Figure.15: Top in Detected Objects in Memes

**3.3.4 Visual-BERT Based Classifier**

Visual-BERT integrates BERT, Faster-RCNN (an object detection model) to deal with multimodal tasks. Visual Bert contains a set of Transformer layers which implicitly align elements in a piece of input text with areas in a related input image by using self-attention techniques [15].

The textual input of the Visual-BERT based classifier is similar to the BERT based classifier. Each sub word has three embedding layers.

电脑屏幕的照片

低可信度描述已自动生成

Figure.16: Visual-BERT textual input illustration

Retrieved from: <https://arxiv.org/pdf/1810.04805.pdf>

Compared with 3 embedding layers in the BERT model, Visual-BERT adds a visual feature embedding layer and introduces special [IMG] token and corresponding token embedding by fusing the output of the model Faster-RCNN. For the segment embedding layer, Visual-BERT adds a new class C to embed the image feature [15].

We implemented Visual-BERT with 2 types of input set: [text, image], [text, tag, image]. Comparing the performance of the model with these two input sets, we can validate the feasibility of the tag extraction procedure in our proposed model. The structure of the Visual-BERT with [text, tag, image] is shown in Figure 17.

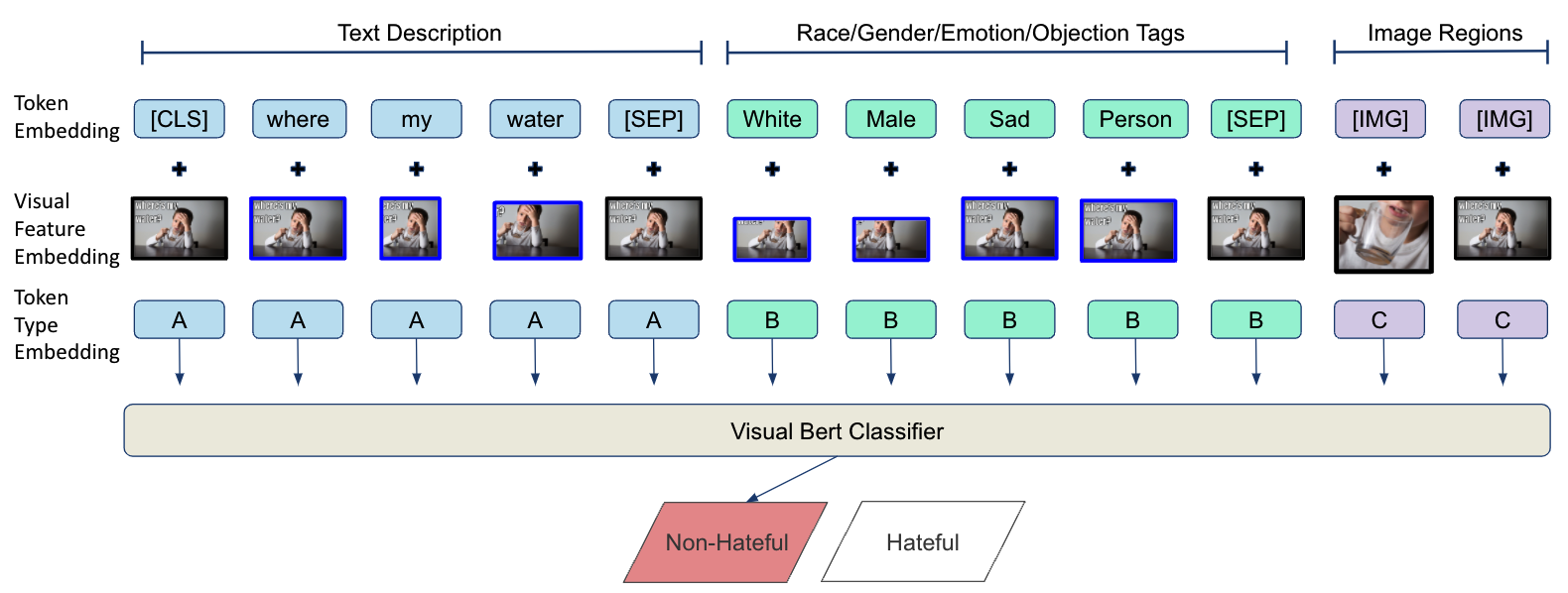
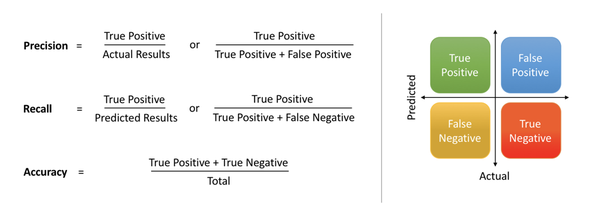


Figure.17: Visual-BERT textual input illustration

We use ‘bert-base-uncased’ data tokenizer which was introduced in the milestone report 2 to do the text tokenization and use the pretrained ‘vit-base-patch16-224-in21k’ model to extract vision features. We use the pre-trained Visual-BERT model ‘uclanlp/visualbert-nlvr2-coco-pre’ as the base model for the finetuning process.

In the model finetuning process, we train the model 50 epochs with batch size 24. In the training process, we set the weight decay (add a penalty term to weights and bias) into 0.01 and set the learning rate at . For every 250 steps, we record the training loss, validation loss, accuracy and auroc score.

**4. Performance evaluation**

We use Accuracy, Precision Recall and F1 measurement to evaluate the model performance [24]. The formula of each measurement score is shown in the figure below.

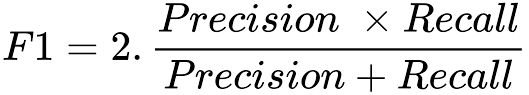


Figure.18: Formula for the model measurement score

We also use the area under the Receiver Operating Characteristic curve (AUROC) to evaluate the results. For the ROC curve, the samples were sorted according to the predicted values of the model and were predicted one after another as positive examples in the sorted order [25]. The values of True Positive Rate (TPR) and False Positive Rate (FPR) were calculated every time, and they were plotted as x and y axes respectively. The formula for TPR and FPR is shown below.

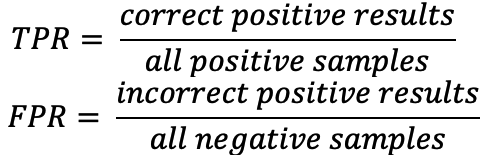


Figure.19: Formula for TPR and FPR

We want the probability that the classifier predicts a sample of true category 1 to be higher than the probability that the sample of true category 0 predicts a sample of true category 1. So that when AUROC is larger, the model performs better in the classification task.

The results of our proposed unimodal model and multimodal model are shown in the below table.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model | Accuracy | Recall | Precision | F1 |
| TF-IDF + LR | 61.8% | 18.2% | 63.3% | 0.28 |
| TF-IDF + SVM | 61.3% | 16.8% | 61.6% | 0.26 |
| TF-IDF + NB | 60.0% | 13.2% | 57.1% | 0.21 |
| TF-IDF + RF | 58.7% | 0.1% | 100.0% | 0.02 |
| TF-IDF + KNN | 58.7% | 57.1% | 0.3% | 0.001 |
| BERT | 58.3% | 35.5% | 63.6% | 0.46 |
| Text-CNN | 63.1% | 45.7% | 50.4% | 0.48 |
| ResNet | 64.5% | 0% | 64.5% | 0 |
| Visual BERT | 58.1% | 42.2% | 50.3% | 0.46 |
| Visual BERT  with tags | 67.3% | 58.8% | 66.0% | 0.62 |

Table.1: Results of the models

Note: The blue one is the baseline model and the red one is the final model we proposed.

Comparing the results of our proposed multimodal model with the baseline model, the accuracy of the classification results has increased from 58.1% to 67.3%, which increases nearly 20%. Meanwhile, the F1 measurement increased from 0.46 to 0.62. This significant improvement validates that compared with the baseline model, our model extracts more useful information from the meme and thus achieves better performance.

Comparing the results of our proposed multimodal model with various unimodal models we implemented. It can be found that the measurement of accuracy and F1-score have been improved significantly. To sum up, among the models constructed in this paper, the multimodal model has better performance than the unimodal model which is in line with our initial expectations.

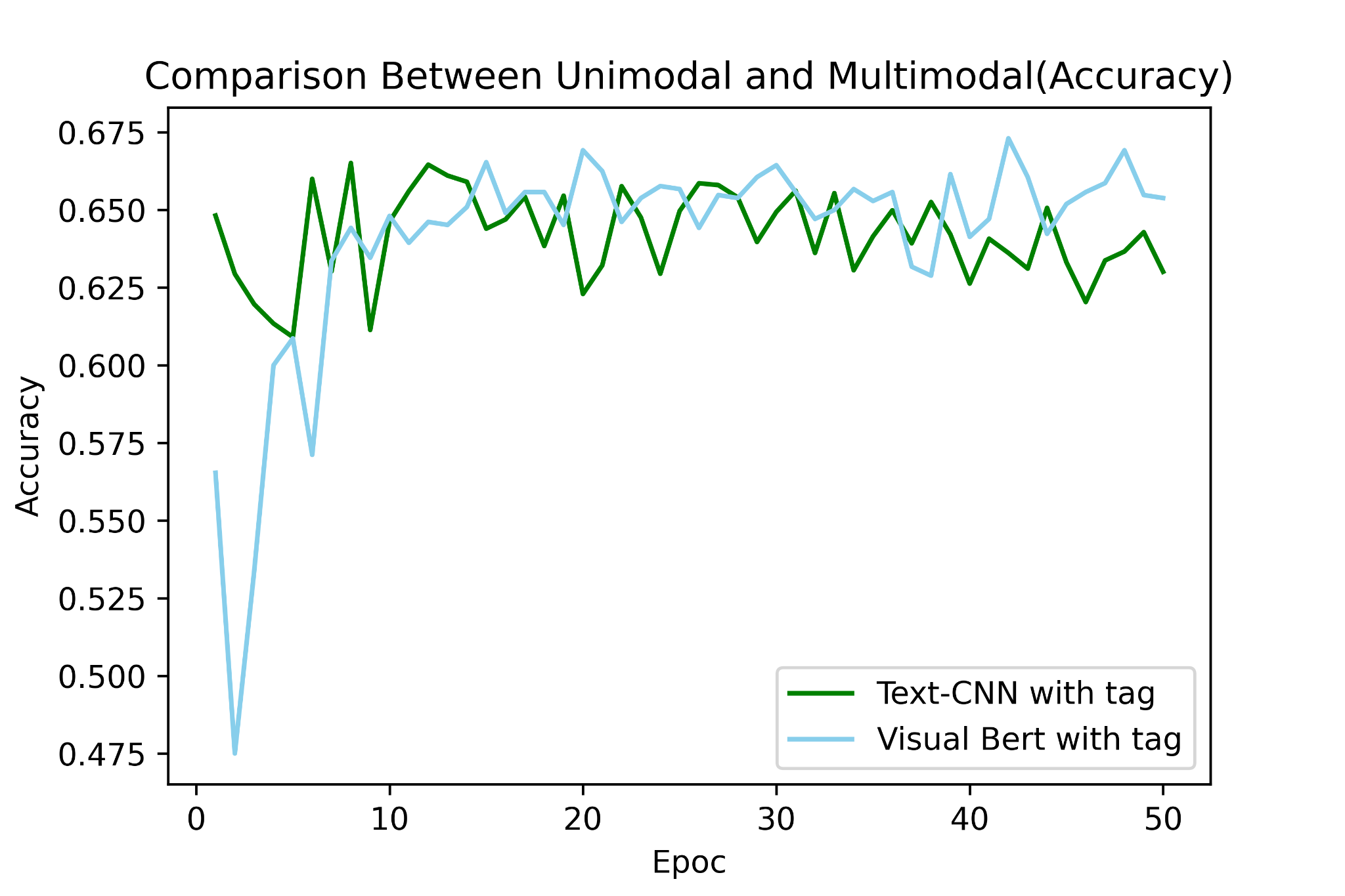
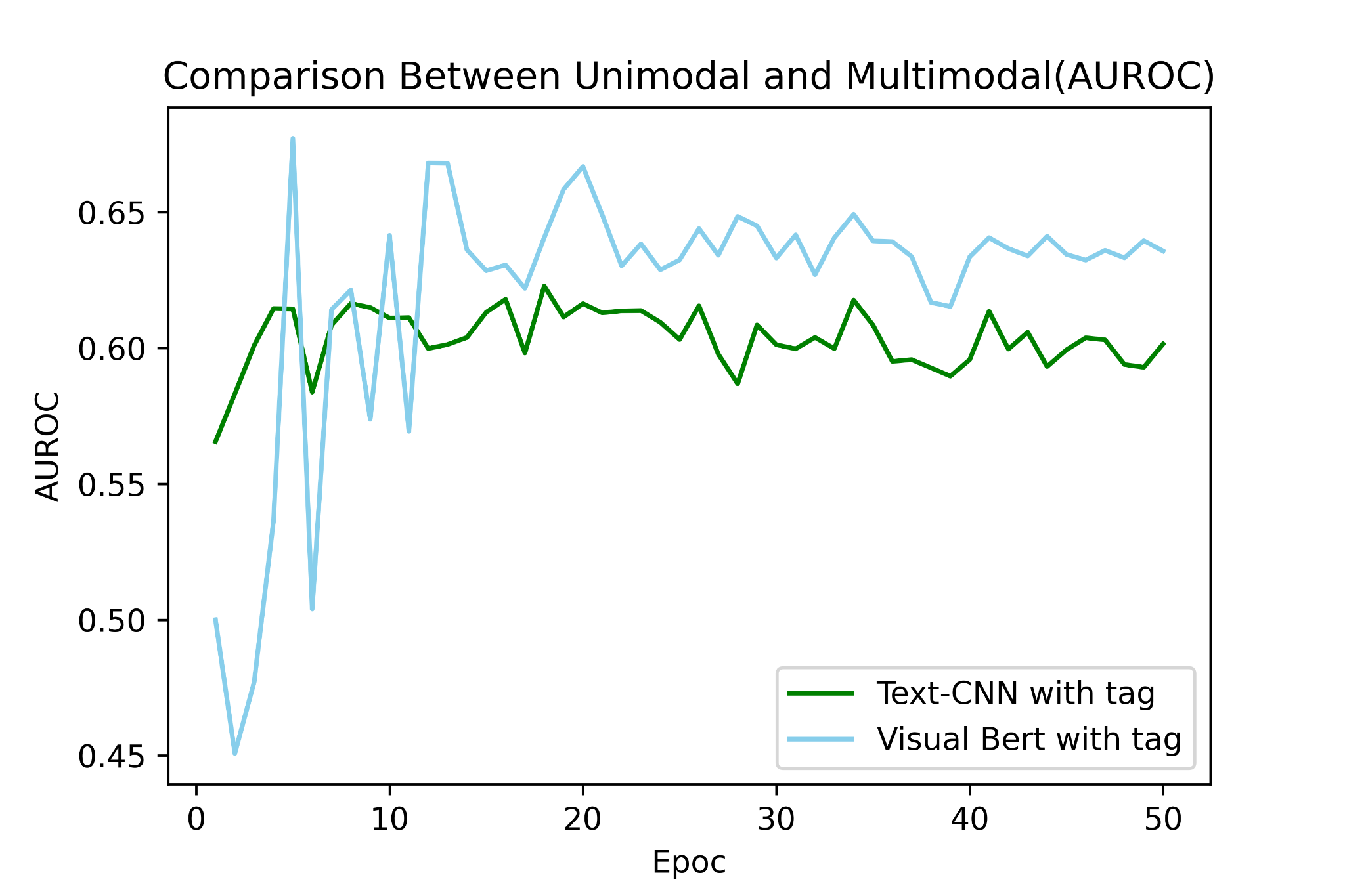
****

Figure.20: Model measurement between unimodal and multimodal

To compare the results more intuitively between the proposed final model and unimodal model with the best performance. Accuracy and AUROC versus epoch number is plotted in Figure.20. We can see from the results, as the number of epochs increases, the performance of the proposed final model stably outperforms that of the unimodal model.

# 5. Conclusion and future work

In this project, we proposed different kinds of classification models for hateful memes, including text-based unimodal, image-based unimodal and text-image-based multimodal models. However, all these models have performance that is not that satisfactory. Therefore, we proposed a new model that first extracted more labels including objects, human races, genders, and emotions to expand our feature vector. The performance increases in terms of accuracy, recall, precision, F1 and auroc. This indicates that our attempts to add more tags to construct our new features make sense. However, currently our labels are still not representative, especially for objects. We could train more specific models that could detect more important objects instead of using pretrained models by others.

Although finding more representative tags could have effects, the project itself has its limitations. According to the initiator of the project, even manual classification of these memes could only achieve an accuracy of 84% [3]. So, unlike traditional image or text classification tasks, meme detection includes some more complicated logic reasoning that is hard for machines to detect. What we attempt to do is to try to find some patterns shared by those logic reasoning that could be captured.

In the future, we will improve the model in the following three aspects:

1. Better tuning the model and implementing other multimodal models as the backbone models. Using the current network structure and hyperparameter set encountered the overfitting problem. We will try to add more batch normalization layers and increase the dropout rate to fix the problem.
2. Extract better hate-related tags. The current face detection model is pretrained on other datasets and we could not evaluate the performance of it. In the future, we could manually label the face type and emotion for our meme dataset to get tags with more accuracy. Besides, the current results of the object detection model are less than satisfactory. We will implement other models such as Google Entity Detection model to get objects with more detailed description [26]. For entity detection, it will try to find corresponding scenarios for each image instead of individual objects.
3. Conduct more pre-processing techniques on the images. In the current meme, the text is shown above the image and may influence the feature extraction procedures of the image. Thus, removing the text from the image may increase the model performance. Besides, currently, there may be different images in a meme. Dealing multiple images at one shot could lead to inaccuracy of our results. Therefore, we should first segment them and deal with them separately.

Via all these newly proposed methods, we hope that the detection model could have a more satisfactory result and could be applied to real life one day.

**References**

1. Chetty, Naganna, and Sreejith Alathur. "Hate speech review in the context of online social networks." Aggression and violent behavior 40 (2018): 108-118.
2. Pranesh, Raj Ratn and Ambesh Shekhar. “MemeSem:A Multi-modal Framework for Sentimental Analysis of Meme via Transfer Learning.” (2020).
3. Kiela, Douwe, et al. "The Hateful Memes Challenge: Detecting Hate Speech in Multimodal Memes.", (2020).
4. Hate speech. Transparency Center. Retrieved March 11, 2022, from <https://transparency.fb.com/zh-cn/policies/community-standards/hate-speech/>
5. “TF–IDF.” Wikipedia, Wikimedia Foundation, March 11,. 2022, https://en.wikipedia.org/wiki/Tf%E2%80%93idf.

[6] Zhang, Ye, and Byron Wallace. "A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification." arXiv preprint arXiv:1510.03820 (2015).

[7] Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).

[8] He, Kaiming, et al. “Deep Residual Learning for Image Recognition.” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, https://doi.org/10.1109/cvpr.2016.90.

[9] Tai, Ying, Jian Yang, and Xiaoming Liu. "Image super-resolution via deep recursive residual network." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

[10] Redmon, Joseph, et al. “You Only Look Once: Unified, Real-Time Object. Detection.” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, https://doi.org/10.1109/cvpr.2016.91.

[11] Panning, A., et al. “Facial Expression Recognition Based on Haar-Like Feature Detection.” Pattern Recognition and Image Analysis, vol. 18, no.3, 2008, pp. 447-452.

[12] AlkaSaliss. (n.d.). Alkasaliss/Democlassi: Democlassi stands for demographic (age, gender, race) and emotions (happy, sad, angry, ...) classification from face images, using deep learning. GitHub. Retrieved March 11, 2022, from <https://github.com/AlkaSaliss/DEmoClassi>

[13] Giannopoulos, Panagiotis, Isidoros Perikos, and Ioannis Hatzilygeroudis. "Deep learning approaches for facial emotion recognition: A case study on FER-2013." Advances in hybridization of intelligent methods. Springer, Cham, 2018. 1-16.

[14] Das, Abhijit, Antitza Dantcheva, and Francois Bremond. "Mitigating bias in gender, age and ethnicity classification: a multi-task convolution neural network approach." Proceedings of the european conference on computer vision (eccv) workshops. 2018.

[15] Li, Liunian Harold, et al. "Visualbert: A simple and performant baseline for vision and language." arXiv preprint arXiv:1908.03557 (2019).

[16] Cecillon, Noé, et al. "Abusive language detection in online conversations by combining content-and graph-based features." Frontiers in Big Data 2 (2019): 8.

[17] Douwe Kiela, Suvrat Bhooshan, Hamed Firooz, and Davide Testuggine. Supervised multimodal transformers for classifying images and text. arXiv preprint arXiv:1909.02950, 2019.

[18] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In Advances in neural information processing systems, pages 91–99, 2015.

[19] Aggarwal, Apeksha, et al. “Two-Way Feature Extraction Using Sequential and Multimodal Approach for Hateful Meme Classification.” *Complexity*, vol. 2021, 2021, pp. 1–7., https://doi.org/10.1155/2021/5510253.

[20] Su, Weijie, et al. "Vl-bert: Pre-training of generic visual-linguistic representations." arXiv preprint arXiv:1908.08530 (2019).

[21] Zhang, W. et al. “Hateful Memes Detection via Complementary Visual and Linguistic Networks.” ArXiv abs/2012.04977 (2020): n. pag.

[22] Schroff, Florian, et al. “FaceNet: A Unified Embedding for Face Recognition and Clustering.” 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, https://doi.org/10.1109/cvpr.2015.7298682.

[23] Lin, Tsung-Yi, et al. “Microsoft Coco: Common Objects in Context.” Computer Vision – ECCV 2014, 2014, pp. 740–755., <https://doi.org/10.1007/978-3-319-10602-1_48>.

[24] Goutte, Cyril, and Eric Gaussier. "A probabilistic interpretation of precision, recall and F-score, with implication for evaluation." European conference on information retrieval. Springer, Berlin, Heidelberg, 2005.

[25] Fawcett, Tom. "An introduction to ROC analysis." Pattern recognition letters 27.8 (2006): 861-874.

[26] Uyar, Ahmet, and Farouk Musa Aliyu. "Evaluating search features of Google Knowledge Graph and Bing Satori: entity types, list searches and query interfaces." Online Information Review (2015).