Saarland University

Department of Computer Science

Master Thesis

Offensive content detection in scarce data and in multimodal setting

Submitted by:

Nikhil Chilwant

Matriculation no.: 2577689

Reviewers:

Prof. Dr. Dietrich Klakow

Dr. Volha Petukhova

20.11.2021

Erklärung

Ich erkläre hiermit, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Statement

I hereby confirm that I have written this thesis on my own and that I have not used any other media or materials than the ones referred to in this thesis

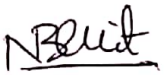
Einverständniserklärung

Ich bin damit einverstanden, dass meine (bestandene) Arbeit in beiden Versionen in die Bibliothek der Informatik aufgenommen und damit veröffentlicht wird.

Declaration of Consent

I agree to make both versions of my thesis (with a passing grade) accessible to the public by having them added to the library of the Computer Science Department.

Saarbrücken, 20.11.2021
(Datum/Date)


(Unterschrift/Signature)

Erklärung

Ich erkläre hiermit, dass die vorliegende Arbeit mit der elektronischen Version übereinstimmt.

Statement

I hereby confirm the congruence of the contents of the printed data and the electronic version of the thesis.

Saarbrücken, 20.11.2021 (Datum/Date)

(Unterschrift / Signature)

Abstract

Internet platform startup companies need to implement a scalable solution for offensive content detection. Social media giants like Facebook have developed their AI-based solution using the billion-dollar investment. In this work, we try to solve the same problem for a startup in a resource-constrained environment. We solve the text-only offensive content detection using a zero-shot learning-based approach. It performs domain adaptation and data selection using the BERT model. We improve the model by transferring learning from multiple datasets using the multi-task learning approach. This approach performed well and gave 88% accuracy and 0.95 AUROC score on the fairly balanced test data provided by Eternio GmbH. We further extend the problem to the 'hateful meme detection' using the recently published Facebook hateful meme challenge dataset. We first analyze the dataset and survey the top-performing approaches to highlight how this problem is challenging and different from the previous ones. The similarity of the problem with the Visual Question Answering (VQA) domain motivated us to try some VQA based models.

Contents

1 Introduction

The problem of offensive content detection has been an area of interest for the Natural Language Processing (NLP) researcher community. It has been a problem of interest from the commercial point of view as well. Given the increase in the scale of the usage of internet platforms, it is essential to find a scalable solution for this problem. For example, Facebook claims to have removed 22.1 million pieces of content in Q3 2020 in the hate speech category only. The AI technology made possible by the billion-dollar investment helped to identify the 95% of the hate speech content [Rosen (2020)]. It is not possible to invest such an enormous amount for a small or medium-sized company. So, in this Master thesis, we try to solve the offensive content detection problem for a startup from the Saarland University - Eternio GmbH - with the limited available resources. We extend Eternio's problem further to make it close to real-life and more challenging. This document provides details about the problem statement, literature review and finally describes how we solved the problem.

1.1 Problem statement

The Eternio GmbH team has developed a web platform to help people to remember their deceased loved ones. The platform allows you to create a web page just like a profile page on a social media platforms (figure 1). The web page has multiple functionalities such as commenting, lightening a candle, schedule a remembrance event etc. The comment feature is relevant to this report. The problem is challenging as there is no dataset with offensive comments for deceased people.

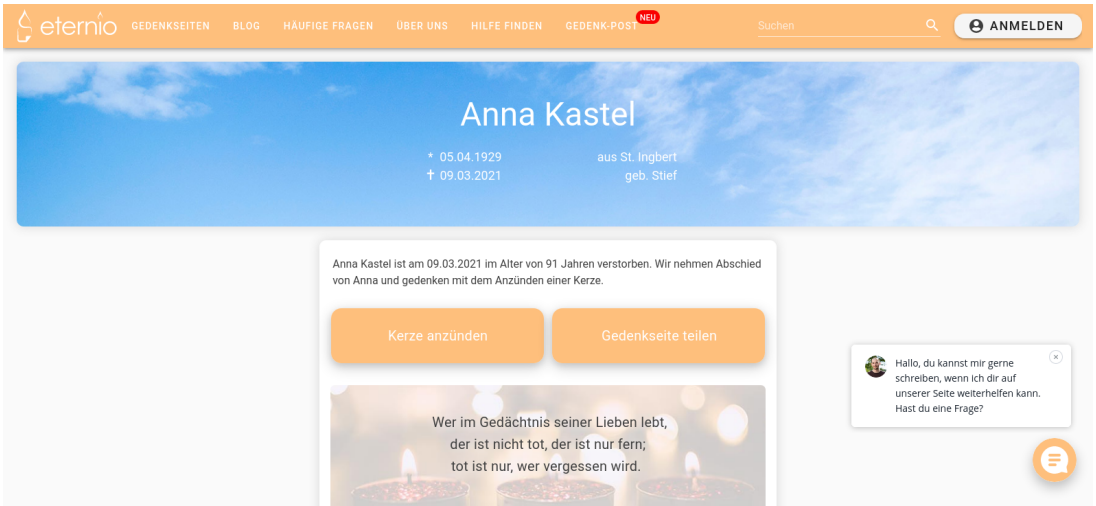


Figure 1: A screenshot of the Eternio web page.

There are many possible types of comments. Below is a list of three comments which shows complexity of the problem. They all show inappropriate emotion on the occasion of death using sarcasm. The first part of the thesis was concerned with developing a

offensive content detector for such text only offensive comments. We refer to this first part of the problem statement as the 'Eternio problem'.

1. Ob er sich Covid jetzt in Rom oder Barcelona geholt hat Verdient hatte ers so oder so *(Whether he got Covid in Rome or Barcelona, he deserved it one way or another)*

2. Jetzt hat der Tod ihn an der Backe Und wir sind ihn zum Glück los *(Now he is dead, we are happy to get rid of him)*

3. Traumhaft eine zukunft ohne ihn *(A dream future without him)*.

The second part of the problem statement is an extension of the above-mentioned 'text only offensive comments'. In the second part, we tried to solve the problem of the detection of hateful memes. A meme is multimedia content that is normally based on an image with text caption embedded in image pixels with the intention of humour [Sabat et al. (2019)]. The recent 'hateful meme challenge' by Facebook has accelerated research in this area. So, we plan to attempt the English version of the problem.

The meme in the figure 2 tries to show why detecting a meme is more challenging than text-based offensive comments. If one considers only the image or only the text separately, it is not offensive content. However, it is offensive when we put them together.

Figure 2: An example of an offensive meme. The lady in the picture is judge Ruth Bader Ginsburg who died in 2020

As you might have noticed, this section did not discuss the terminology in detail and used the terms 'offensive' and 'hateful' interchangeably. Section 3 discusses the terminology in detail.

2 Preliminary experiments

Before we moved on to do the literature review on how to solve the data unavailability problem, it was necessary to check the performance of the currently available approaches. The transformer-based BERT model is the current best performing model for the NLP problems in general. So, we tried to train it on three selected datasets. Table 1 gives a brief overview of the datasets. The datasets were selected because they had a good variety of comments and were close to the real life comments. For example, GermEval 2018 dataset was created by collecting selected tweets and both Jigsaw datasets were created using Wikipedia comments.

Dataset name	Language	Task	Size
GermEval 2018	German	Classify tweets into offensive and non-offensive categories. The OFFENSE category covered abusive language, insults, as well as merely profane statements.	8541
Jigsaw toxic comment	English	Classify the Wikipedia comments into 'toxic', 'severe toxic', 'obscene', 'threat', 'insult' and 'identity hate'.	223,549
Jigsaw unintended bias	English	Build a model that recognizes toxicity and minimizes the unintended bias with respect to mentions of identities.	1,971,916

Table 1: A brief overview of the GermEval2018 [Wiegand (2019)], Jigsaw toxic comment[Jigsaw (2018)] and Jigsaw unintended bias datasets [Jigsaw (2019)]. The 'size' of dataset counts train, test and dev splits.

Since the thesis's goal is limited to identifying the offensive contents, we modified the labels from the datasets accordingly. Table 2 shows the performance numbers for the BERT based models. We selected the pre-trained BERT language model according to the dataset.

Dataset name	Accuracy(%)	AUROC
GermEval 2018	97.38	0.98
Jigsaw toxic comment	95.4	0.974
Jigsaw unintended bias	99.9	0.999

Table 2: Performance numbers for the datasets. These numbers imply that BERT is good at detecting offensive comments.

Next, the BERT model developed for the GermEval 2018 dataset was used for classifying the test dataset given by Eternio GmbH. The test dataset had a total of 480 comments with 268 offensive and 212 non-offensive comments. Table 3 shows the performance num-

bers. The model identifies all comments as non-offensive. We need to find a method to train the model in order to incorporate the definition of the offensive comments from the target dataset. So, the next step was to explore the literature related to the few-shot learning.

These experiments helped in two ways. First, it gave hands-on practice required for the thesis. Second, it confirmed that the pre-trained German BERT is capable of solving the offensive comment detection task.

	Precision	Recall	F1-score	Support
non-offensive	0.44	1.00	0.61	212
offensive	0.00	0.00	0.00	268
accuracy			**0.44**	480
macro avg	0.22	0.50	0.31	480
weighted avg	0.20	0.44	0.27	480

Table 3: Performance of the BERT based model trained on the Eternio test dataset. The recall value of offensive comment is zero, implying the failure of our approach.

3 Literature review

This section provides details about the literature reviewed in order to solve the problem. It covers all background concepts relevant to the upcoming sections. It discusses the terminology used in this report and gives a brief overview of the machine learning concepts mentioned in this report. It is impossible to discuss them in detail, and so, the section provides references where the reader can read further. Also, it assumes the reader to have an elementary background in machine learning.

3.1 Terminology

The researchers have coined various terms depending on the context, e.g. abusive, hostile, cyberbullying, hate speech, insulting, profane, malicious intent, othering language etc. [Schmidt and Wiegand (2017)] This document uses the term 'Offensive' in general and includes all the terms used in this context by researchers.

We define the term 'offensive content' by extending the official community guideline definition given by Facebook Inc.:

A comment indicating the death of a person as a positive event is considered as offensive. Additionally, a direct or indirect attack on people based on characteristics, including ethnicity, race, nationality, immigration status, religion, caste, sex, gender identity, sexual orientation, and disability or disease. We define attack as violent or dehumanizing (comparing people to non-human things, e.g. animals) speech, statements of inferiority, and calls for exclusion or segregation. Mocking hate crime is also considered offensive [Kiela et al. (2020)].

It is important to survey the approaches proposed for detecting offensive content. Below is the summary list of the learning approaches for detecting the text-only offensive content [Schmidt and Wiegand (2017)]:

1. Word-based and character-based N-gram approach.
2. Word generalization: carry out word clustering and represent sets of words as generalized features.
3. Use the fact that an offensive comment usually has a negative sentiment.
4. Word list-based approach.
5. Use POS-information enriched tokens.
6. Knowledge base approach to include context.
7. Meta information based appraoch e.g. use user information.

The above list is not exhaustive, but it should give the reader an idea about the research's pre-deep learning direction. Today, deep learning based approaches are popular but above approaches could offer enhancement in performance.

3.2 Few shot learning

Few shot learning (FSL) is a popular machine learning technique in a resource constrained environment which aims at obtaining a good performance with limited labeled data points i.e. supervised information given in the training set. The formal definition goes as this: "for a learning task, FSL deals with a data set $D = \{D_{train}, D_{test}\}$ consisting of a training set $D_{train} = \{(x_i, y_i)\}_{i=1}^{I}$ and testing set $D_{test} = \{x^{test}\}$. Here, the I is small. If $p(x, y)$ is the joint probability distribution with input x and output y, FSL aims to find the optimal hypothesis by approximating withing the hypothesis space" [Wang et al. (2020)]. Wang et al. (2020) have provided an excellent overview of the taxonomy and mathematical intuition behind FSL algorithms.

FSL becomes a 'one shot learning' paradigm when there is only one example in the supervised information for the task. If there are no examples in the supervised information for the task, it is called as a 'zero shot learning' paradigm [Wang et al. (2020)].

Transfer learning is one of the popular techniques in the FSL domain where knowledge from the source domain/task with abundant training data is transferred to train the model for data-scarce target domain/task. Pan and Yang (2009) have provided an excellent overview of the transfer learning taxonomy. 'Domain adaptation' is a type of a transfer learning. Usually, a standard machine learning paradigm assumes source/target task and domain to be the same. However, when source/target tasks are the same but domains are different, it becomes a 'domain adaptation' problem. This difference is the domain distributions is called as the 'domain shift' [Csurka (2017)]. This domain shift between the Eternio test dataset and GermEval 2018 occurs as the offensive comments are coming from the different contexts. Csurka (2017) has given a good introduction to domain adaptation.

There are multiple ways of measuring domain shift. For example, the Kullback-Leibler (KL) divergence is a parametric way of measuring the domain shift as it requires density estimation. The Maximum Mean Discrepancy (MMD) is the non-parametric criterion for comparing distributions. Since the estimation of densities is a non trivial task, non-parametric criterions are desired [Pan et al. (2010)]. The MMD (d_k) between the probability distributions P & Q in the reproducing kernel Hilbert space (H_k) with kernel k is given by [Ma et al. (2019)]

$$d_k^2(P, Q) := ||\mathbf{E}_P[x] - \mathbf{E}_Q[x]||_{H_k}^2 \tag{1}$$

Now, we discuss the core mathematical intuition behind FSL techniques. We provide brief

details relevant to understand the logic. Please refer Wang et al. (2020) for the detailed discussion. The intuition is derived from the error decomposition of the total error. We first introduce a few terms required to understand the concept.

The expected risk (R) for a given hypothesis (h) with respect to the joint probability distribution $p(x, y)$ is given by

$$R(h) = \int l(h(x), y)\, dp(x, y) = \mathbb{E}[l(h(x), y)] \tag{2}$$

Since $p(x, y)$ is unknown, we estimate it with empirical risk which is average of sample losses over the training set D_{train} of I samples.

$$R_I(h) = \frac{1}{I} \sum_{i=1}^{I} l(h(x_i), y_i) \tag{3}$$

Now, we define a few symbols necessary to understand the figures. We denote $\hat{h}$ be the function that minimizes expected risk in the equation 2 i.e. $\hat{h} = \text{argmin}_h\, R(h)$. Since $\hat{h}$ is unknown, we approximate it by some $h \in \mathcal{H}$. We define $h^* = \text{argmin}_{h \in \mathcal{H}}\, R(h)$ be the function in $\mathcal{H}$ which minimizes the expected risk. h^* is the best approximation of $\mathcal{H}$. Finally, we define h_I be the function in $\mathcal{H}$ that minimizes empirical risk i.e. $h_I = \text{argmin}_{h \in \mathcal{H}}\, R_I(h)$. h_I is the best hypothesis obtained by the empirical risk minimization.

Using the above mentioned terms, we define decompose the total error as

$$\mathbb{E}[R(h_I) - R(\hat{h})] = \mathbb{E}[R(h^*) - R(\hat{h})] - \mathbb{E}[R(h_I) - R(h^*)] \tag{4}$$

The first part on the right hand side of the equation is called as the 'approximation error' and the second part is called as the 'estimation error'. The approximation error gives idea about how close the optimal hypothesis h^* in $\mathcal{H}$ is close to the optimal hypothesis h. The empirical error gives idea about estimation about how close h_I is to h in $\mathcal{H}$. This is shown in the figure 3. One can reduce the estimation error by increasing the number of training samples i.e. I. In the case of FSL, the I is small which results in the empirical risk $R_I(h)$ being far from the good estimate of the expected risk $R(h)$. This is the core issue of the FSL which makes it harder to solve. The comparison with a supervised learning problem with sufficient number of training samples is shown in the figure 3.

To solve the problem of unreliable estimation of h_I, various techniques are used. These techniques solve the problem by including prior knowledge to enhance either data, model or the algorithm aspect. We discuss them briefly here.

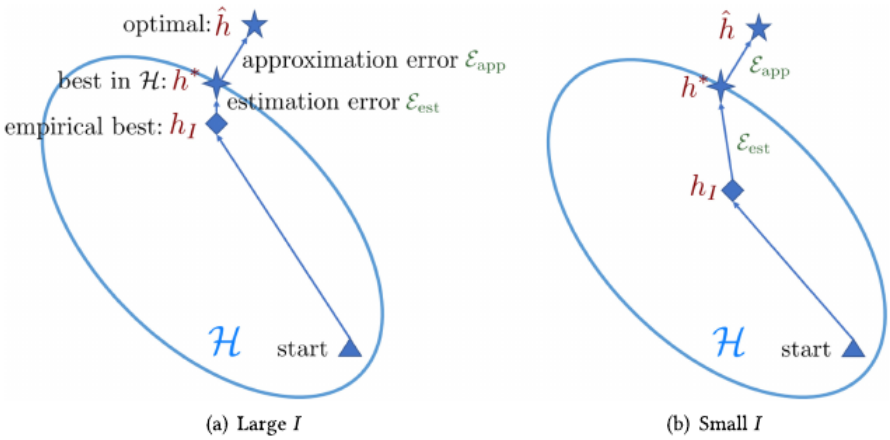


Figure 3: Comparison of learning methods with different sample sizes [Wang et al. (2020)].

The first technique called data augmentation involves increasing the sample size in the training dataset D_{train} to a sufficiently large number. Then one can use the standardized machine learning algorithms and models to solve the problem of finding reliable estimate for empirical risk minimizer h_I. The data can be augmented by transforming samples from the training dataset or from the similar large sized dataset.

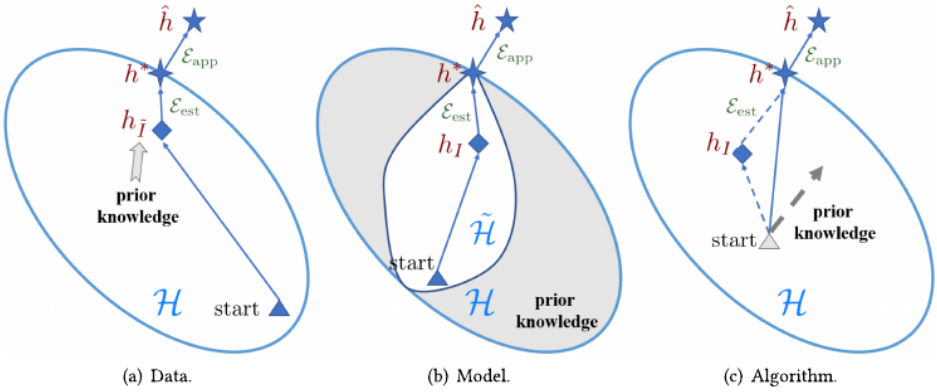


Figure 4: Different approaches to solve the FSL problem [Wang et al. (2020)].

The second technique regards to the model. In this technique, prior knowledge is used to reduce the complexity of $\mathcal{H}$ which results into smaller space $\tilde{\mathcal{H}}$. The prior knowledge can be acquired by multiple ways. For example, the multi-task learning technique learns from multiple tasks. It learns task-generic and task-specific information. We discuss in details about it in the section 3.6. In embedding learning, the hypothesis space is reduced by projecting data points onto lower dimensional space. The other methods use extracted knowledge from the training data (learning from external memory) or they generate data points (generative modeling).

The third technique pertains to the algorithm. Algorithm decides the search strategy

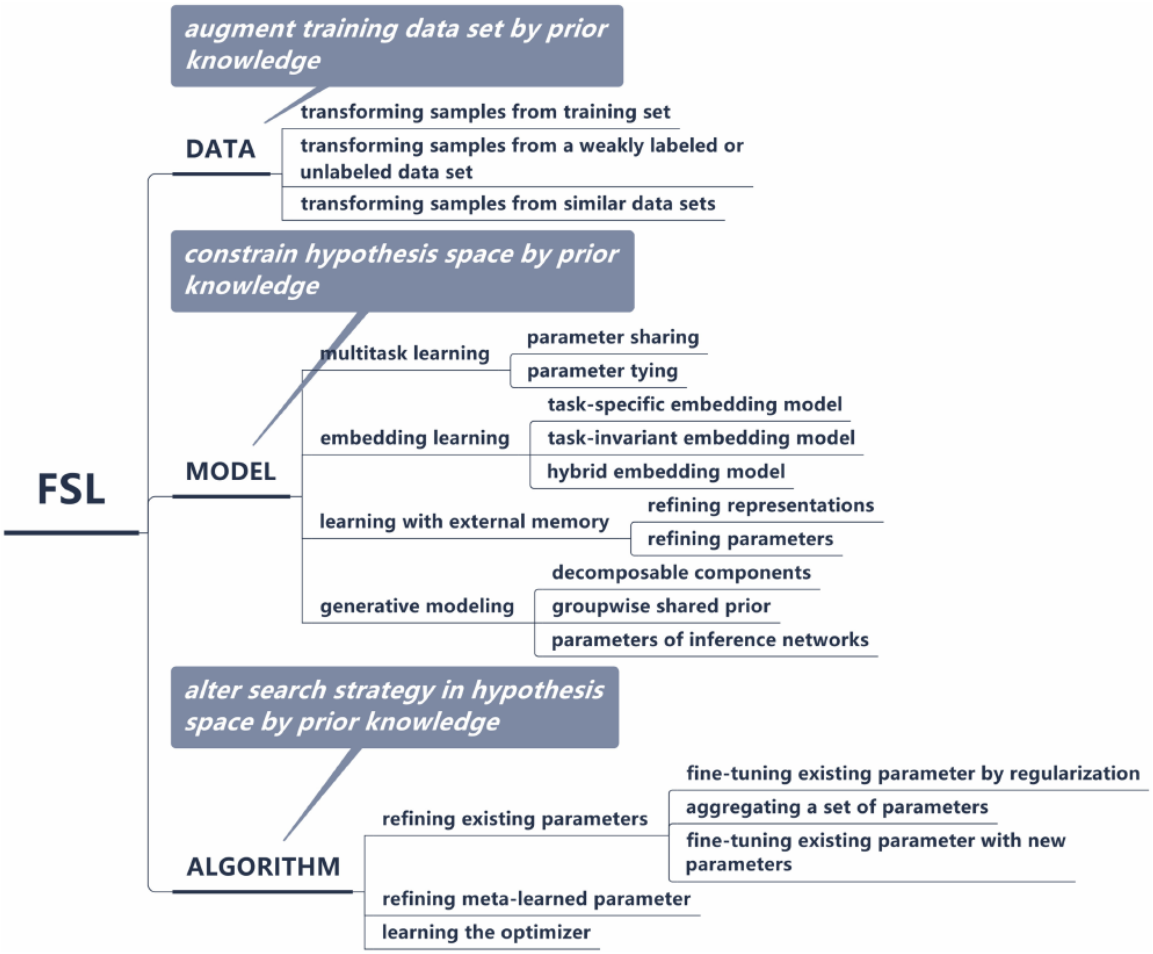


Figure 5: The detailed taxonomy of strategies applied to solve the unreliable estimation of empirical risk [Wang et al. (2020)].

to reach the best hypothesis h^* in the hypothesis space $\mathcal{H}$. This achieved by applying strategies for better parameters search and devising better optimiser.

The figure 5 gives further details about the taxonomy of the FSL strategies. Wang et al. (2020) has given thorough overview of taxonomy.

3.3 t-SNE

In this work we will need a data visualization technique to visualize our datasets. t-distributed Stochastic Neighbour Embedding (t-SNE) is a popular technique to visualize datasets. Methods like Principle Component Analysis (PCA) are popularly used for dimensionality reduction to preserve significant structure of the high dimensional data in the low dimensional space. Van der Maaten and Hinton (2008) argued that the t-SNE capture the local structure of the high dimensional data while revealing the global structures like presence of clusters. This subsection briefly discusses the intuition behind t-SNE. We recommond to refer Van der Maaten and Hinton (2008) for the detailed discussion.

The idea of t-SNE was developed on Stochastic Neighbour Embedding (SNE) given by Hinton and Roweis (2002). The key idea is to convert the Euclidean distances between the data points in high dimensional space into conditional probabilities. So, if we have two data points x_i and x_j in the high dimensional space, the conditional probability $p_{j|i}$ is the probability that the point x_j will be selected if we select x_i. Here, Hinton and Roweis (2002) assumed that the probability density function is Gaussian with x_i at the center with reasonable variance of σ such that it vanishes for distant neighbours. The mathematical representation of $p_{j|i}$ is given below. Note that $p_{i|i} = 0$.

$$p_{j|i} = \frac{exp(\frac{-||x_j - x_i||^2}{2\sigma_i^2})}{\sum_{k \neq i} exp(\frac{-||x_i - x_k||^2}{2\sigma_i^2})} \tag{5}$$

The data points in the low dimensional space are denoted by y_i. Hinton and Roweis (2002) set the variance as $\frac{1}{\sqrt{2}}$. So, the conditional probability distribution for the data points in the low dimensional space is given by

$$q_{j|i} = \frac{exp(-||y_j - y_i||^2)}{\sum_{k \neq i} exp(-||y_i - y_k||^2)} \tag{6}$$

The conditional probabilities p_i and q_i should have minimum mismatch. The mismatch is measure by Kullback-Leibler divergence which is minimized using the gradient descent method. The cost function that would be minimized is given by

$$\sum_i KL(P_i||Q_i) = \sum_i \sum_j p_{j|i} \log \frac{p_{j|i}}{q_{j|i}} \tag{7}$$

SNE develops on the above equations. The SNE approach had a few drawbacks which are addressed by t-SNE. The t-SNE has two main differences from SNE. First, it uses the STudent t-distribution instead of Gaussian distribution for computing the similarity between the low dimensional space. Second, it simplified the cost function with symmetric version and simpler gradient.

In Van der Maaten and Hinton (2008); Van der Maaten (2013) the author has explained all mathematical details in excellent way. For the purpose of this work, we used t-SNE mainly to visualize various datasets and understand the domain shift. As shown in the figure 6 t-SNE does the job of clustering similar data points and visualizing the data very well. We also recommend to refer Wattenberg et al. (2016) to understand how to tune the t-SNE hyperparameters.

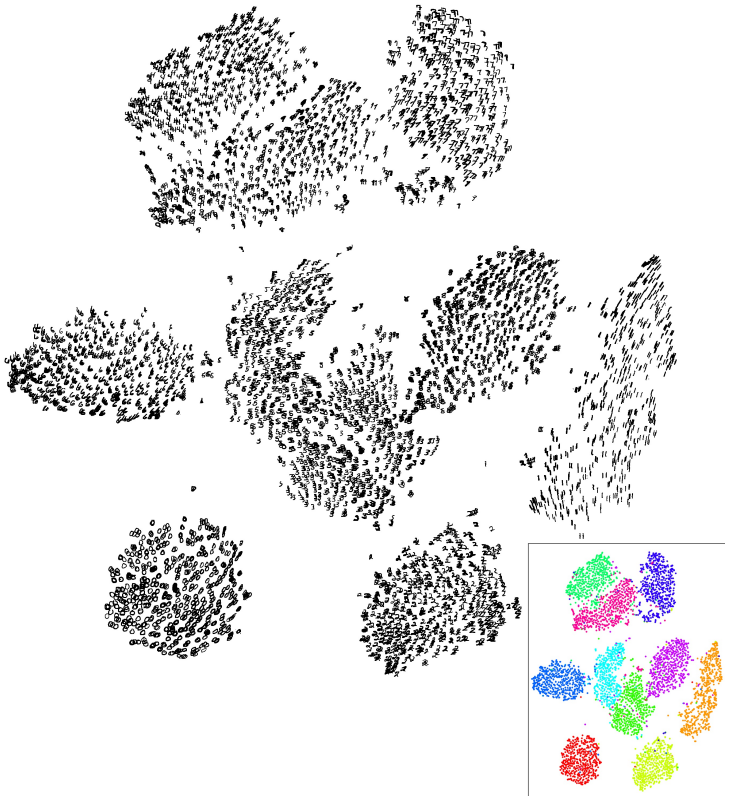


Figure 6: Visualization of MNIST dataset using t-SNE shows us a good clustering [Van der Maaten and Hinton (2008)].

3.4 Multi-modal offensive content detection

The second problem, hateful meme detection, can be approached using the multi-modal machine learning (MML) technique. A meme has data in the form of two forms: text and image, i.e. two modalities. The hateful meme classification problem requires the model to learn from both modalities. It involves steps like representation of modalities, translation to map them, alignment to semantically link them, fusion to join them for prediction and co-learning to transfer knowledge from modalities [Baltrušaitis et al. (2018)]. Baltrušaitis et al. (2018) has provided a good overview of the MML area. The remaining subsection tries to give an overview of the approaches proposed for detecting hateful memes.

Yang et al. (2019) have explored various fusion and attention techniques and reported significant performance improvement compared to simple concatenation. They found concatenation with symmetric gated fusion with Sparsemax activation function and deep clone attention mode gives the best performance. They used ROC-AUC as the criteria of evaluation as it provides model performance across all scoring points. Unfortunately, they haven't shared the used dataset. Gomez et al. (2020) have created MMHS150K dataset with 150 K tweets containing hateful content. They present multiple approaches to fuse textual and image data using feature concatenation, spatial concatenation and textual kernel model. However, they concluded that these multi-modal approaches do not

outperform text-based classification approaches. They mentioned that the reasons behind such a poor performance are noisy data, the complexity of the problem and scarcity of the memes in the dataset. The Facebook hateful meme dataset solves these issues. We provide more details about the dataset in section 5.

To detect a meme as offensive, one needs to have social and cultural context about the subject it talks about. Vijayaraghavan et al. (2021) have proposed a late fusion-based approach for detecting offensive multi-modal content. They extracted semantic features from text using a word embedding and character embedding. Further, they extract social and cultural context features using information about the content creator and his social media network. Fusing these two resulted in a better performance which was verified using the interpretability methods. However, this approach assumes the availability of the up-to-date author information database, which may not be the case every time.

One important problem that multiple papers have pointed out is that there is no standardized offensive meme dataset. As a result, the performance numbers for the models are not comparable. The work by Sabat et al. (2019) found that a classifier based on only images performs better than the text-only classifier. The multi-modal classifier outperforms both but image features dominate the classification. They attribute this to the nature of the dataset, higher dimensionality of the image features than the text features and modal capacity for them. On the other hand, other studies found textual features dominating the classification [Gomez et al. (2020)]. As a result, one needs to select a 'truly' multi-modal for this task.

3.5 Visual question answering

The 'Visual question answering' (VQA) task comes under the domain of Vision and Language representation learning. The domain of VLR learning includes other tasks like Visual reasoning but we are going to focus only on VQA in this document. Further, we are going to limit ourselves to only on static images instead of videos. The requirements of the hateful meme classification, which is a multimodal classification problem, are similar to the Visual Question Answering (VQA) problem [Gomez et al. (2020)]. Hence, it is helpful to review the VQA literature.

A VQA system takes an image and an open-ended, free-form natural language question about the image as input and produces a natural language answer to it as an output [Antol et al. (2015)]. The field has evolved rapidly in the last decade. The figure 7 shows a timeline of introduction of datasets. We briefly discuss the motivation behind introduction of these datasets.

Antol et al. (2015) introduced the VQA 1.0 dataset by including 'realistic' images from MS COCO dataset. Their work is important because they were the first one to define the

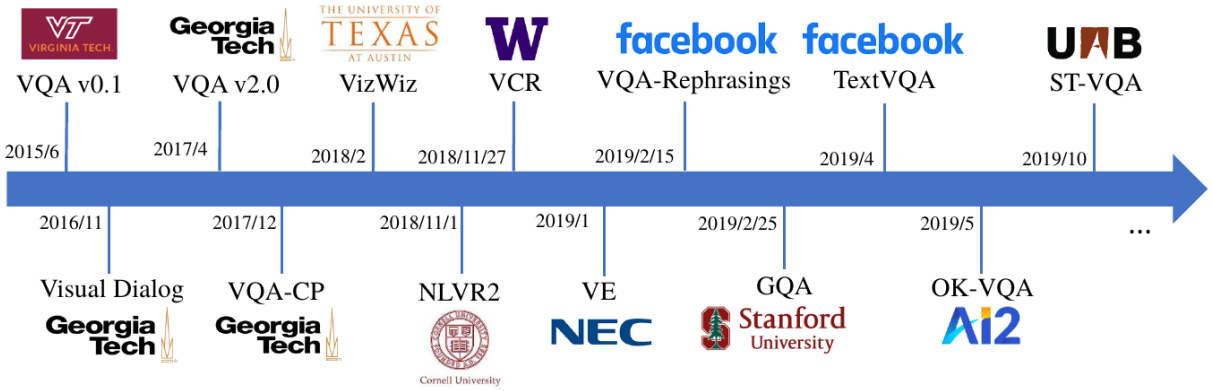


Figure 7: A timeline of recent important datasets introduced in the VQA field [Gan (2020)].

VQA task. Next, the Vision Dialog dataset was introduced by Das et al. (2017) to define the Visual Dialog task. The Visual Dialog task includes and image, a history of dialogues consisting of questions and answers. In the end, we ask a follow up question to machine and expect an answer in the free-form natural language.

In 2017, VQA 2.0 was introduced by Goyal et al. (2017) to address a few short comings of VQA 1.0. They balanced influence of language priors by introducing more images in the dataset. IN the same year, Agrawal et al. (2018) from the same research group from Georgia Technology pointed out a few more shortcomings with VQA v1.0 and VQA v2.0 datasets. They proposed Grounded Visual Question Answering (GVQA) model that had inductive biases and constraints in the architecture itself that prevents introduction of priors. Additionally, they gave new splits for the VQA (v1.0 and v2.0) datasets and called it VQA Changing Priors (VQA-CP). Gurari et al. (2018) introduced VizWiz dataset and decided to take the VQA task closer to the real scenario. The dataset included real life challenging images with questions that may not be answered sometimes. They were motivated by a genuine desire to help blind people with VQA research.

In 2018, Natural Language Visual Reasoning for real (NLVR2) dataset was introduced Suhr et al. (2019). The dataset contained English sentences paired with photographs. The task was to decide whether the text aligns with the photos. Shortly after, a group from Washington university decided to stretch the boundaries of the field with the (Visual Commonsense Reasoning) VCR dataset. The task required 'understanding' the scene to answer the question and explaining the rationale behind the answer. The Visual Entailment task introduced by Xie et al. (2019) also had image-sentence pairs where the premise is defined by an image and the goal is to predict if the image semantically entails the text.

While a group of researchers were pushing forward the VQA models in complexity aspect, a few others were focusing on the adversarial aspect. Shah et al. (2019) introduced the

VQA-Rephrasings dataset as well as a model agnostic framework to make them robust.

Hudson and Manning (2019) pointed out three shortcomings of VQA datasets published before. First, the datasets have a strong and prevalent real-world priors throughout the data. Second, previous datasets had questions which were easy to answer for VQA models under the influence of real-world priors. GQA attempted to solve that. Next, Singh et al. (2019) pointed out that previous datasets rarely ask questions related to the text in the image. So, they introduced TextVQA dataset and the Look, Read, Reason and Answer approach.

Most of the datasets discussed till now asked questions related to the presence or count of the objects in the image. They did not require reasoning or outside knowledge. The task of knowledge based visual question answering is closer to the real-world scenario. Hence, Marino et al. (2019) introduced the Outside-Knowledge VQA (OK-VQA) dataset. It included question that cannot be answered by image alone as they require the outside knowledge. For example, the figure 8 shows one of such task.

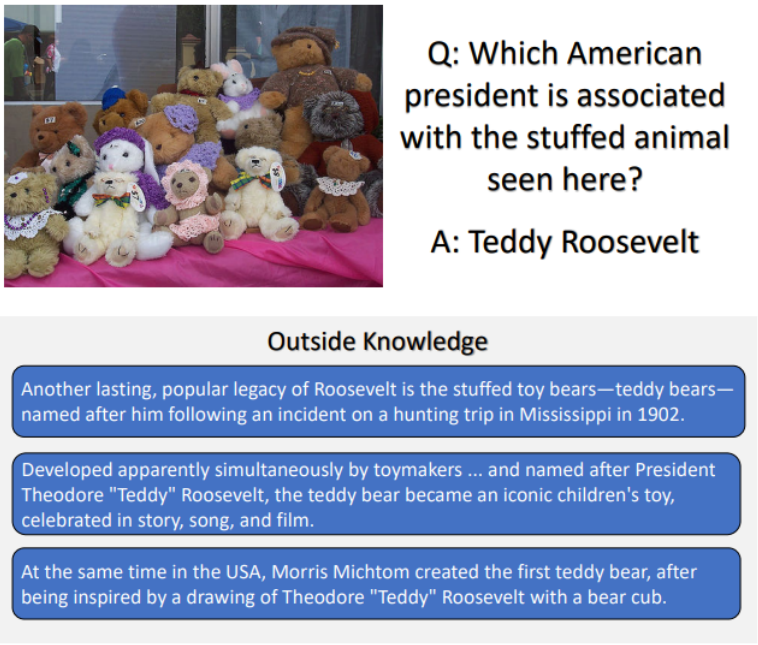


Figure 8: An example of outside knowledge visual question answering task [Marino et al. (2019)].

Overall, the field is rapidly evolving and new datasets, architectures are still being introduced every year. We try to use some of them when we discuss the hateful meme detection part of the work.

3.6 Multi-task learning

Generally, a machine learning model is trained by finding optimal parameters for a single task. Multi-task learning generalises over related tasks by leveraging the domain-specific information from the related tasks. One can view this as an inductive transfer which introduces inductive bias to improve the model performance. Ruder (2017) has given an excellent overview of the MTL concept. This report discusses only introductory concepts of the MTL.

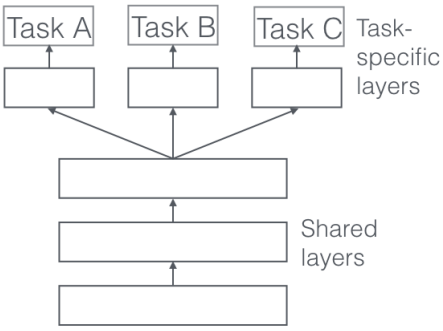


Figure 9: A schematic overview of the MTL by hard parameter sharing [Ruder (2017)].

There are two ways of implementing MTL in Deep Learning: 'hard' and 'soft' parameter sharing. Caruana (1997) proposed the MTL by hard parameter sharing and remains a popular approach even today. As shown in the figure 9, it involves sharing specific layers across the tasks and having task-specific last layers. This approach is different from the transfer learning approach as it involves training the model for tasks simultaneously [Zhang and Yang (2021)]. Ruder (2017); Zhang and Yang (2021) have discussed approaches for task selection. Its discussion is beyond the scope of this report. A generic algorithm to implement MTL is given in the figure 10.

Ruder (2017) gave the intuition behind why the MTL approach works. First, the inclusion of different tasks helps out average the noise patterns that are generally present in every task. This 'implicit data augmentation' results in a better model. Second, the MTL can help to focus on relevant features during training. Third, the model can learn useful features for a task from other tasks efficiently. Ruder (2017) calls it 'eavesdropping'. Finally, MTL helps to find a better-generalised model by introducing the inductive bias and representation bias.

3.7 Hateful meme detection

This subsection first explains why the dataset published by Facebook is different from previously published datasets. Then, it discusses the winning approach from the hateful meme detection challenge along with required background in pre-trained models.

Initialize model parameters Θ randomly.
Pre-train the shared layers (i.e., the lexicon encoder and the transformer encoder).
Set the max number of epoch: $epoch_{max}$.
//Prepare the data for T tasks.
for t *in* $1, 2, ..., T$ **do**
  Pack the dataset t into mini-batch: D_t.
end
for $epoch$ *in* $1, 2, ..., epoch_{max}$ **do**
  1. Merge all the datasets:
  $D = D_1 \cup D_2 ... \cup D_T$
  2. Shuffle D
  for b_t *in* D **do**
    //b_t is a mini-batch of task t.
    3. Compute loss : $L(\Theta)$
      $L(\Theta)$ = Eq. 6 for classification
      $L(\Theta)$ = Eq. 7 for regression
      $L(\Theta)$ = Eq. 8 for ranking
    4. Compute gradient: $\nabla(\Theta)$
    5. Update model: $\Theta = \Theta - \epsilon\nabla(\Theta)$
  end
end

Figure 10: The generic MTL algorithm [Liu et al. (2019)].

3.7.1 The hateful meme challenge dataset

The hateful meme challenge document given by Kiela et al. (2020) points out that previously published datasets are not 'truly' multimodal. That is, during the training phase of the model, either text features or vision features dominate. The hateful meme challenge dataset overcomes this by including 'benign confounders'. Figure 11 shows an example of such a confounder. In that context, if one changes the image of the animal to the rose, it becomes harmless. Similarly, if the text is modified and the image of the animal is kept the same, it again becomes benign. However, the original image is hateful. As a result, the model biased towards visual or textual features will not perform well on this dataset.

Figure 11: An example of a benign confounder. It is possible to flip the category of the meme by changing either just the image or the text [Kiela et al. (2020)].

Kiela et al. (2020) observed that models pre-trained in multimodal fashion perform better. The performance of models is given in the table 4. For the performance comparison, they use AUROC as the parameter.

Rank	Proposed by	AUROC	Accuracy
1	Zhu	0.8450	0.7320
2	Muennighoff	0.8310	0.6950
3	Velioglu Rose	0.8108	0.7650
4	Lippe et al.	0.8053	0.7385
5	Sandulescu	0.7943	0.7430

Table 4: The hateful meme challenge ranking dashboard

3.7.2 Relevant important models and architectures

This subsection gives brief introduction to some important pre-trained models and fusion techniques used for hateful meme detection approach. We assume that the reader is familiar with the workings of the BERT [Devlin et al. (2019)]. For good summary of important models in visio-linguistic domain, please refer to the work of Mogadala et al. (2021).

VL-BERT [Su et al. (2019)]

In the initial phase of development of pre-trained generic visiolinguistic models, task specific models were designed using ad-hoc combination of features derived from off-the-shelf computer vision and NLP models. VL-BERT was a generic single cross-modal transformer architecture based model. The authors claim that the model has better generalization linguistic and visual features. The architecture of the model is modification of the BERT architecture. It takes in visual features in the form of Region of Interests in images and sub-words as linguistic features from input sentences. Su et al. (2019) demonstrated that the model can be fine tuned for visual-lingustic tasks like VQA, VCR (Visual Commonsense Reasoning) and grounding referring expression [Mao et al. (2016)] tasks.

UNITER [Chen et al. (2020)]

UNiversal Image-TExt Representation (UNITER) learns joint multimodal embeddings using the four pre-training tasks: (1) Masked Language Modeling (MLM) which masks word from sentence but keeps image embedding intact, (2) Masked Region Modeling (MRM) which keeps word embeddings intact and hides a portion of image embedding, (3) Image-Text Matching (ITM) which learns instance level matching between image and text (4) Word-Region Alignment (WRA) which uses Optimal Transport algorithm and learns alignment between word embedding and image tokens. The architecture is shown in the figure 12. Since it is pre-trained on fine grained level of image and text, UNITER is important for hateful meme detection problem.

ERNIE-ViL [Yu et al. (2020)]

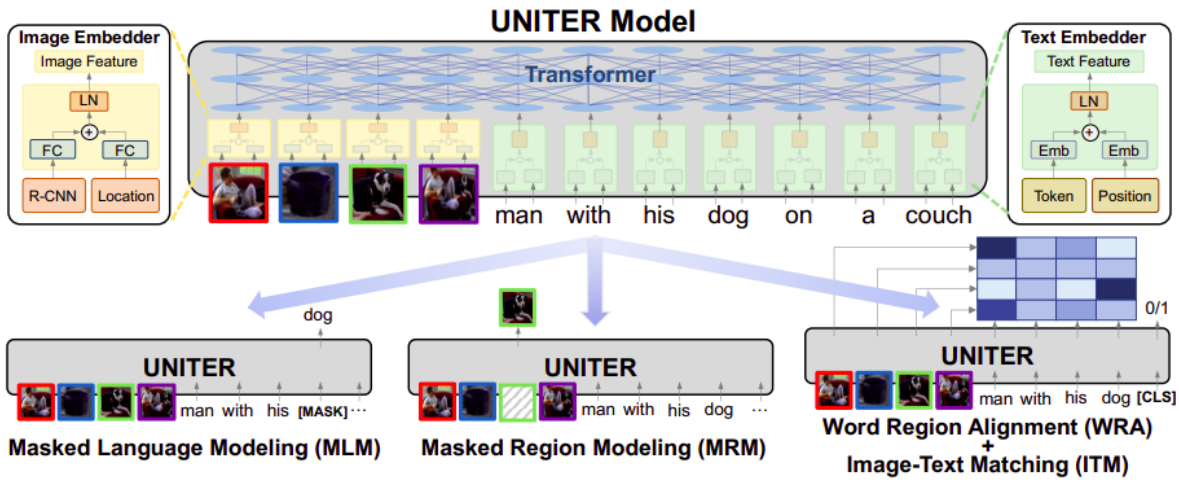


Figure 12: Architecture of the UNITER model depicting pre-training tasks [Chen et al. (2020)].

The before discussed models use visual grounding tasks to pre-train models and therefore do not capture the semantic alignment across vision and language. The ERNIE-ViL which was inspired by ERNIE uses structured knowledge from scene graphs. Pre-training the model on the scene graph prediction task results into learning about semantics from the scene e.g. object prediction, attribute prediction and relationship prediction. The figure 13 shows that the detected region of the image and token sequence of the text are passed to the two stream cross-modal transformer. Using the Scene Graph Parser, Object Prediction, Attribute Prediction and Relationship Prediction tasks are constructed to learn cross-modal semantic alignment.

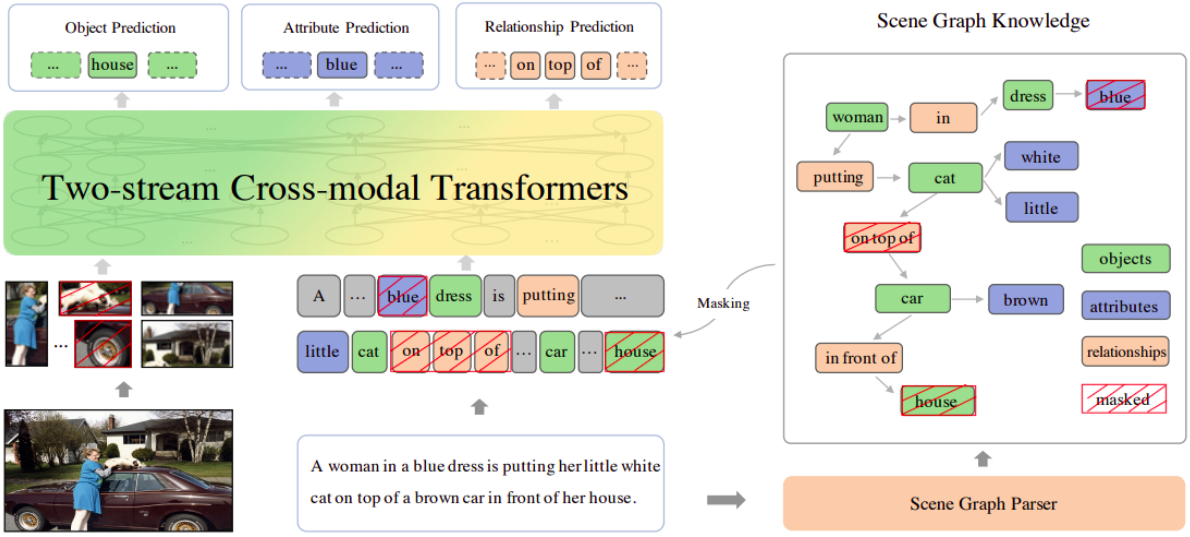


Figure 13: Overview of ERNIE-ViL pre-training to learn semantic alignment [Yu et al. (2020)].

Vision-and-Language Large-scale Adversarial training (VILLA) [Gan et al. (2020)]

Recently, adverserial training method has shown a great potential in creating a good generalized pre-trained models. Gan et al. (2020) developed VILLA using this idea. They introduced perturbations in the embedding space rather than directly into the images and text inputs.

Relation-aware Graph Attention Network (ReGAT) [Li et al. (2019)]

Most VQA models prior to ReGAT used the strategy of learning a multimodal joint representation of images and questions. For example, visual features extracted from Convolutional Neural Network (CNN) or Region-based CNN (R-CNN) and RNN based question encoding are fused together. In the end, a joint representation is learned to pass it through the answer predictor. This framework works for the VQA task but there exist a semantic gap between image and the natural language. ReGAT tried to address this by introducing relation encoders which capture inter-object relations. It differs in comparison to other VQA models as it uses question adaptive inter-object relations. This property can make ReGAT useful for the hateful meme detection problem.

The ReGAT model considers two types of visual relationships : explicit relations and implicit relations. Explicit relations are pre-defined semantic and spatial relations between objects learned using datasets like Visual Genome [Krishna et al. (2017)]. In the end, the graph is constructed based on pre-defined relations. In the case of implicit relations, relations are captured from the input image to model the interactions between detected objects. Figure 14 shows examples of spatial and semantic relation. ReGAT uses Graph Attention Network (GAT) for capturing explicit relations and implicit relations are captured with question adaptive graph by filtering out relations irrelevant to the question. The figure 15 gives overview of the architecture.

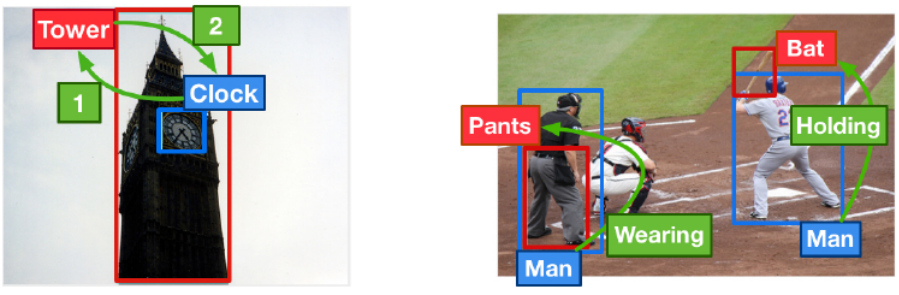


Figure 14: Examples of spatial relations (left) and semantic relations (right). The green arrows show relation, the labels in the green boxes are the relation labels and the labels in the red boxes are the class labels of the objects [Li et al. (2019)].

MUTAN [Ben-Younes et al. (2017)]

Bilinear model approach is a powerful approach for VQA problems because they encode full second order interaction. One important disadvantage with the bilinear approach is

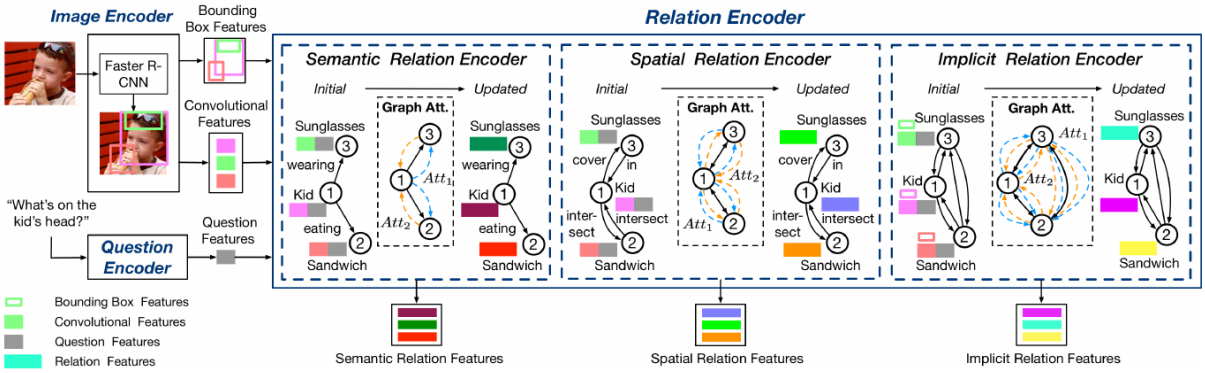


Figure 15: An overview of the architecture of ReGAT. The set of object regions and question features are fed into relation encoders to learn relation aware question adaptive visual features. In the end, they are fused with question representation to predict the answer. Multimodal fusion and answer predictor are not shown for simplicity. [Li et al. (2019)]

very high number parameters. The MUTAN architecture solves this problem using the Tucker decomposition. It also gives interpretable repatriation of learnable parameters. The figure 16 gives a brief overview of the MUTAN fusion scheme.

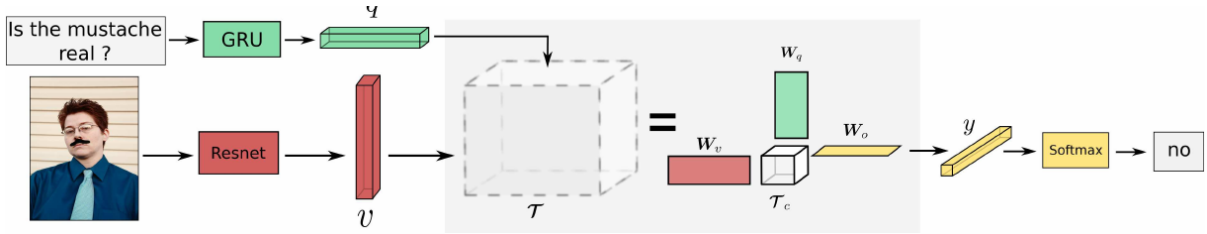


Figure 16: An overview of the MUTAN fusion scheme for VQA. The bilinear interaction between question and image features is parameterized by $\mathcal{T}$ tensor. The tensor is decomposed using the Tucker decomposition into three intra-modal matrices and a smaller, simpler tensor $\mathcal{T}_c$ [Ben-Younes et al. (2017)].

As shown in the example (figure 17), MUTAN has better attention mechanism. MUTAN performs well on OK-VQA dataset as well.

3.7.3 Approaches

This subsection discusses the top three approaches that performed well in the hateful meme detection competition.

Zhu's approach

This approach given by Zhu (2020) gave AUROC as 0.8450 and accuracy as 73.20%. Zhu (2020) ensembled modified VL-BERT [Su et al. (2019)], UNITER [Chen et al. (2020)], ERNIE-Vil [Yu et al. (2020)] and VILLA [Gan et al. (2020)].

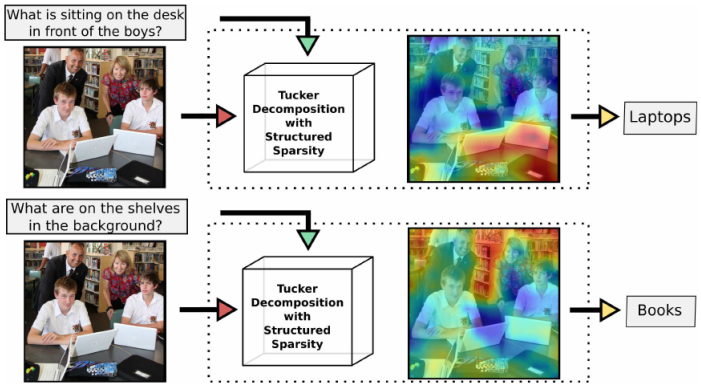


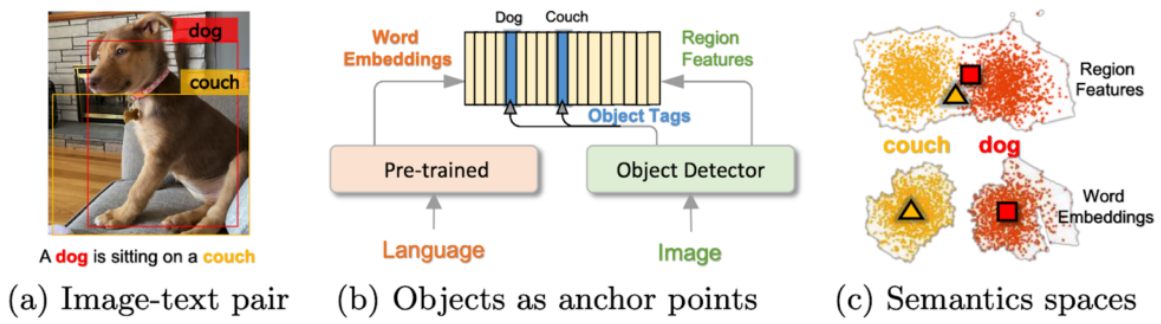
Figure 17: The MUTAN fusion when integrated into attention mechanism gives better results. In figure, we can see that the regions with larger attention score (proportional to intensity of the red colour) indicate good understanding of question content. This enables to perform well in VQA task. [Ben-Younes et al. (2017)]

The first step was to use OCR, remove the text from the meme and do image in-painting. It helps in the next step, collecting additional data sources using the Google web entity detection API and the FairFace classifier. Google web entity detection API gives additional image context, and FairFace gives tags regarding gender and race for the person present in the meme. Next, the VL-BERT was trained by representing all external labels as special types of text tokens and linking them to the special image region using the concept of 'visual feature embedding'. The concept of visual feature embedding is inspired by the OSCAR [Li et al. (2020)]. We use the figure 18 to explain the concept. As shown in the figure 18, given an image and the text pair, we identify 'anchor points', which are 'dog' and 'couch' in this case (figure a). Next, we map them in word embedding space and visual region feature space (figure b). The semantic space in figure c shows that though these anchor points are close to each other in the visual region feature space, they are far away in the word embeddings space. Zhu (2020) used this idea and trained an 'extended' VL-BERT using the caption text, object entity tags, race tags and image regions (figure 19).

Finally, they used the idea that when an image and text do not match contextually, the meme is probably hateful. So, they used UNITER with the image-text matching head as a part of the ensembled model. This idea gave AUROC as 0.845 and accuracy as 73.20%.

The author also tried the below approaches which did not make into the final model:

1. Adapter-transformer [Pfeiffer et al. (2020)] on VL-BERT
2. Adding OSCAR
3. Adding OSCAR fine-tuned with image feature extractor (ResNet101, Faster-RCNN)
4. LXMERT [Tan and Bansal (2019)] fine-tuned with image feature extractor



(a) Image-text pair (b) Objects as anchor points (c) Semantics spaces

Figure 18: (a) An example of input image-text pair. (b) The object tags are used as anchor points to align image regions with word embeddings of pre-trained language models. (c) The word semantic space is more representative than image region features [Li et al. (2020)].

Figure 19: Architecture of extended VL-BERT [Zhu (2020)].

5. Apply entity tag on UNITER and OSCAR
6. Apply super-resolution to image before extract feature
7. Throwing more pretrained data at VL-BERT
8. ROC-Star loss [Yan et al. (2003)]
9. Contextual text augmentation [Kobayashi (2018)]
10. Text only toxic comment classifier combine with VL-BERT
11. VILLA's adversarial fine-tuning [Gan et al. (2020)]
12. Object attribute from the Vision Genome [Krishna et al. (2017)] object detector as feature

Muenninghoff's approach

Muennighoff (2020) enesembled VisualBERT, OSCAR, UNITER, ERNIE-ViL and DeVL-Bert. The approach involved three steps: preparation, modelling and ensembling (figure 20).

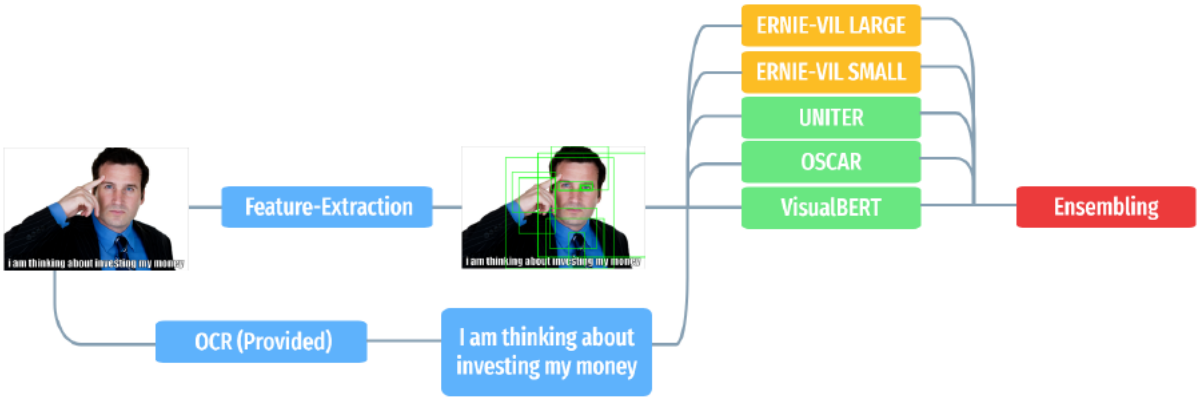


Figure 20: Steps involved in the approach by Muennighoff (2020).

In the first phase of preparation, features were extracted using the Detectron 2 framework [Wu et al. (2019)]. With diverse models trained on different datasets feature extraction has been found to be more effective [Jiang et al. (2018)]. In addition, varying Region of Interest is found to be helpful in improving the performance. In the second step i.e. modelling, the models were fine tuned. Muennighoff (2020) used VisualBERT, OSCAR, UNITER, ERNIE-ViL in this step. The final step was ensembling where each model's performance was averaged with three to five seeds.

It is also important to look at the failed ideas.

1. ROC-Star loss

2. Hyperparameter tuning by changing architecture, parameters.
3. The dataset contains misspelled words like 'niqqa', 'nigga'. Muennighoff (2020) tried to change it to the correct form using the dictionary. This did not help.
4. Pretraining BERT on jigsaw hate speech (text only) did not improve performance.
5. Using features with different min_boxes& max_boxes with padding: No improvement.
6. Label Smoothing [Szegedy et al. (2016); Müller et al. (2019)]
7. Ensemble of ensembles
8. Re-initiating final layers & pooler.
9. Adding new words to BERT.
10. Flagging profanity words with a tag.

Velioglu and Rose's approach

Velioglu and Rose (2020) proposed to expand the dataset with inclusion of memes from the Memotion dataset [Sharma et al. (2020)]. The authors used Memotion dataset [Sharma et al. (2020)] for this purpose and selected 328 memes to grow the training dataset. The 'similar' memes were selected manually and added to the training dataset. The second step was image encoding where 100 boxes of 2048 dimensional region based features were extracted. The visual embeddings were projected into the textual embeddings space and passed through the transformer layers (figure 21). In the third and last step of training, VisualBERT models are fine tuned for various hyperparameters and ensembled.

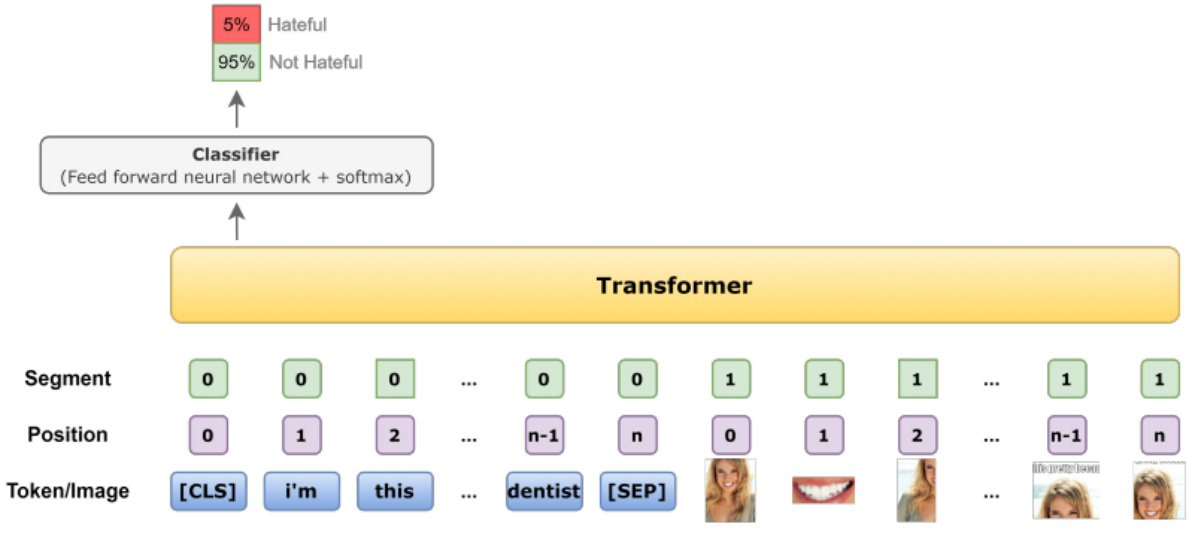


Figure 21: Overview of the architecture used by Velioglu and Rose (2020).

Velioglu and Rose (2020) also tried the below approaches which did not help:

1. Using accuracy metric than Cross-Entropy for training.
2. VisualBERT models which are pre-trained on different datasets.
3. Adding all the samples and annotations from the Memotion dataset to the training set.
4. Different encoders for image feature extraction.

4 Approach and analysis for the offensive text detection

From the preliminary experiment results, we concluded that the model performs poorly because of the 'domain shift'. This section describes the approach to reduce the domain shift in the text-only Eternio offensive content detection problem. It also gives details of the experimental setup used for it. In the end, we share results and analyze them.

4.1 Domain adaptation for the Eternio problem

Pan et al. (2010) classified the domain adaptation problem area as related to the transductive transfer learning setting. In this setting, we have the source domain labels but do not have the target domain labels. The classical machine learning literature recommends the feature representation transfer approach for this setting. In this approach, we encode the knowledge in the form of learned feature representation. With a 'good' learned feature representation of the target domain, the model performance improves. Today, deep learning-based approaches dominate, and we decided to use them.

For solving the text-only Eternio offensive comment detection problem, we used Ma et al. (2019)'s work. Generally, the approaches proposed for domain adaptation are either of the type 'discrepancy based' or 'adversarial based'. Ma et al. (2019) found the adversarial approach difficult to train and unstable. So, they proposed a discrepancy based approach that uses BERT - currently best performing pre-trained language model. Ma et al. (2019) named the approach as 'domain classification with data selection'.

The 'domain classification' step was inspired by the 'curriculum learning', which uses the prior knowledge about the difficulty of the training examples. This step uses the BERT to select data points from the 'source domain' similar to the 'target domain'. The probability scores from the Softmax layer quantifies domain similarity. So, we design a 'curriculum' for training the model consisting of the samples with increasing difficulty levels, i.e. decreasing probability value (higher probability implies 'easy' problem). Figure 22 shows the setup. It shows a modified classification layer on top of the setup given by Devlin et al. (2019).

The 'domain adaptation' step uses the MMD concept. The training objective function was designed as in equation 8. There, Ma et al. (2019) combined the cross-entropy loss (L) and MMD (d_k) with the regularization factor (λ). The regularization factor will be learned during the training. S is the collection of the labelled source domain data, and k is the rational quadratic kernel. The data points in the S are denoted by x_i and y_i. The D_t and D_s are target and source domain datasets.

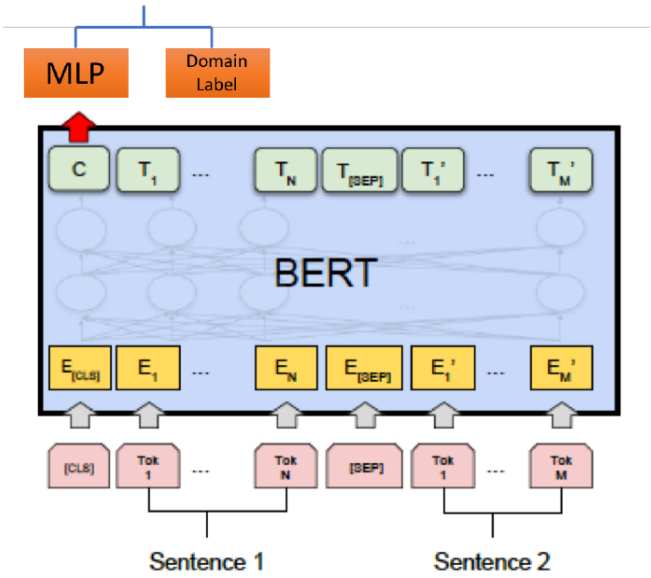


Figure 22: In the BERT domain classification setup, we passed the hidden state of [CLS] token of the input sentence pair to a multi-layer perception [Ma et al. (2019)].

$$\min_{\theta} \frac{1}{|S|} \sum_{x_i, y_i \in S} L(x_i, y_i; \theta) + \lambda . d_k^2(D_s, D_t; \theta) \tag{8}$$

The training setup looks as shown in the figure 23. We obtain the latent representation by collecting the output from the layers before the classification layer of the BERT. Using the latent representations, we calculated the domain loss. The classification loss (cross-entropy loss) was calculated using the predicted and target label.

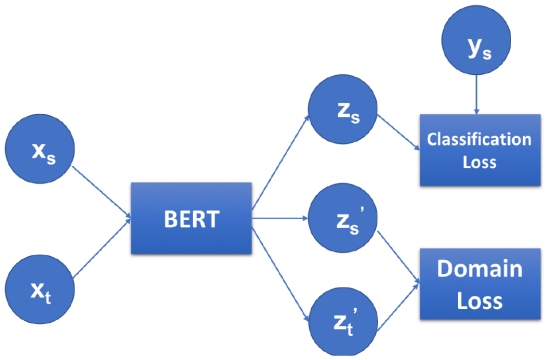


Figure 23: Setup for BERT domain adaptation with MMD-based domain regularization. The loss values are calculated using the latent representations and predicted labels. The notations are as follows : x_s: labeled source data, x_t: unlabeled target data, z_s: predicted label for source data, y_s: target label for the source data, z'_s: latent domain representation of the source data, z'_t: latent domain representation of the target data [Ma et al. (2019)].

To obtain better performance, we incorporated training data points from datasets other than GermEval 2018. Specifically, we evaluated GermEval 2017, GermEval 2019, Filmstarts, IWG and HASOC datasets for this purpose. We also implemented the MTL algorithm given by Liu et al. (2019). The following subsection describes details about how we implemented these steps.

4.2 Experiment details and intermediate observations

This subsection describes details about the selected datasets. It explains why we selected those datasets. Also, the subsection gives implementation details about the BERT domain classifier, domain adaptor and the multi-task learning setup.

4.2.1 Overview of the datasets

[Wiegand (2019)] published the GermEval 2018 dataset as a part of the GermEval shared task evaluation campaign in 2018. The dataset is a collection of tweets from the Twitter platform. Each comment had labelled as 'offensive' and 'other'. We modified it as '1'(offensive) and '0'(non-offensive) for our purpose. Wiegand (2019) have nicely described steps taken to make the dataset free from biases and make it suitable for offensive comment detection. They incorporated an excellent variety of comments by avoiding tweets with commonly used offensive words like 'idiot' or 'Schmarotzer'. Further, they avoided bias towards the current trending topic as tweets are usually heavily influenced by recent events, e.g. tweets related to the refugee crisis were dominant in the tweet stream. They also took care of not including URLs and avoiding re-tweets to be part of the dataset. Finally, the dataset had a good number of offensive tweets (33%). Due to all these reasons, we shortlisted this dataset.

[Wojatzki et al. (2017)] created the GermEval 2017 dataset by collecting customer review comments about 'Deutsche Bahn'. They gathered around 2.5 million documents from internet forums. Since this collection had a lot of irrelevant documents, they refined the data for annotation using the Support Vector Machine method. Since the test dataset, i.e. the target domain had offensive comments that expressed negative sentiment without using offensive words, this dataset was useful. We changed the annotation labels as '1' (offensive) for negative sentiment and '0' (non-offensive) for positive and neutral sentiment for our experiment. The dataset also had a good balance of comments with 26% comments with negative sentiment. All these reasons motivated us to shortlist this dataset.

The GermEval 2019 [Struß et al. (2019)] dataset is created by extending GermEval 2018 task. The dataset was collected on Twitter again, but the authors aimed to increase diversity among offensive comments. In GermEval 2018, they collected tweets randomly with query-term based sampling. For this iteration, they selected users using heuristics. While selecting the tweets, they carefully included a wide variety of political views. So,

in GermEval 2018, the right-wing extremist tweets were dominant. In this iteration, they included a good mixture of political views in the tweets. For example, the dataset included extreme left-wing views (30%), extreme right views (56%), antisemitic views (10%). Further, they included a few more tweets for debiasing (4%). Please refer Struß et al. (2019) for more details. The variety of topics was the main reason why we selected this dataset.

The Filmstarts dataset was created by scraping film reviews in German from filmstarts.de [Guhr et al. (2020)]. The review comments had ratings between 0.5 and 5 stars. The datasets contain a good mixture of positive (72%) and negative reviews (28%). We selected this dataset mainly because we wanted to grow the data points with negative sentiment. The dataset includes all the reviews till 2018.

The Interdisciplinary Working Group of the Department of Computer Science and Applied Cognitive Science, University of Duisburg-Essen, published the IWG dataset [Ross et al. (2017)]. This dataset contained tweets related to the refugee crisis that happened in 2016. The study measured the reliability of annotations. They used ten hashtags to gather offensive tweets related to refugees. Next, they did further processing on the data. Please refer Ross et al. (2017) for more details. We wanted to include the racial, religious aspect in our training dataset. So, we selected this dataset.

[Mandl et al. (2019)] published the HASOC dataset as a part of the Hate Speech and Offensive Content Identification in Indo-European Languages task. We used only the German dataset for our purpose. They extracted the data from social media platforms like Twitter and Facebook. For this, they collected the posts by using shortlisted keywords and hashtags. The dataset had 88% non-offensive comments and 14% offensive comments. We included it to expand our collection of offensive comments.

The target datasets contain a lot of offensive comments which use sarcasm and irony. Unfortunately, we could not find a German sarcasm dataset equivalent to the English dataset like SARC [Khodak et al. (2017)]. Table 5 gives statistics related to the above-described datasets. The figure 24 shows t-SNE visualization of the datasets.

4.3 BERT domain classifier

As explained in the section 4.1, we need to perform the domain classification step to get the value of the probability of belonging to the target dataset. For this purpose, we created a new dataset by pairing each shortlisted dataset with the target dataset, i.e. Eternio dataset. So, we created six such 'paired' datasets e.g. GermEval 2017 and Eternio, GermEval 2018 and Eternio etc. The table 6 gives an idea about the format of the dataset. We assigned the dataset label as 1 for the Eternio dataset comments and 0 for the other comment texts.

Dataset name	Short task description	Size
GermEval 2018 [Wiegand (2019)]	Identification of offensive language	8541
GermEval 2017 [Wojatzki et al. (2017)]	Sentiment classification	26209
GermEval 2019 [Struß et al. (2019)]	Identification of offensive language	7025
Filmstarts [Guhr et al. (2020)]	Movie review sentiment classification	71229
IWG [Ross et al. (2017)]	Identification of offensive comment	469
HASOC [Mandl et al. (2019)]	Offensive Content Identification	3819

Table 5: Details of the datasets selected for multi-task learning step. The 'size' of the dataset is total number of data points in the dataset which includes train, development and test samples.

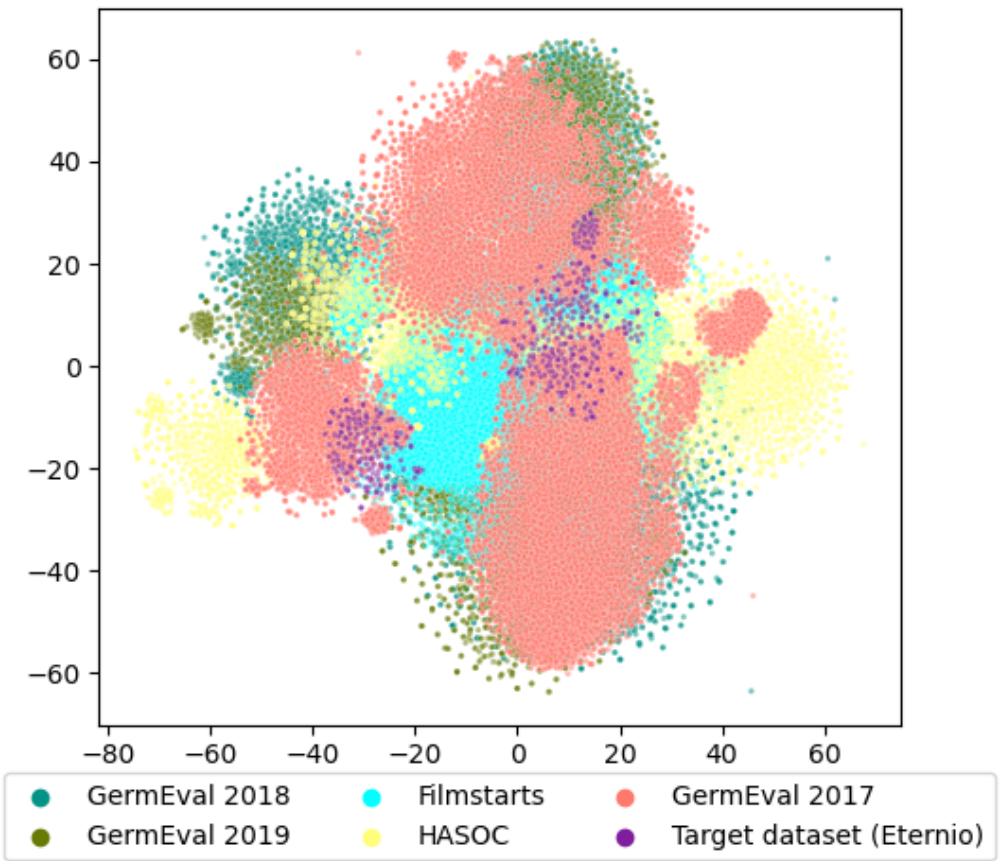


Figure 24: t-SNE visualization of the datasets (perplexity = 50). We observe several clusters for each dataset as the datasets had provided fine level of categorization of comments. For example, GermEval 2017 provides further labelling of offensive comments as 'explicit', 'implicit' and 'other'.

Comment text	toxic_label_max	dataset_label
Bahn ja weil in Wuppertal ...	0	0
...	...	...
loooser	1	1

Table 6: The schema of the combined dataset has comment texts, toxicity labels (toxic_label_max) and the dataset labels (dataset_label). During the domain classification step, we use only the dataset_label.

We trained the domain classifiers and used the classifier to emit the probabilities. The setup used for this purpose was as shown in the figure 22 with one fully connected layer on top of the pre-trained BERT. The table 7 summarizes performance on the ten-fold cross-validation.

Dataset	accuracy (%)	F1 score	ROC AUC
GermEval 2017	99.9	0.997	0.999
GermEval 2018	99.8	0.985	0.999
GermEval 2019	99.9	0.996	0.999
Filmstarts	99.9	0.986	0.999
IWG	99.9	0.988	0.999
HASOC	99.9	0.991	0.999

Table 7: Performance summary of the BERT domain classifier for all shortlisted datasets. The high performance numbers imply that the classifier is trained well. The numbers are attributed to the good discriminative nature of BERT.

The classifier performance was good due to the discriminative nature of the pre-trained BERT model [Ma et al. (2019)]. Once the model emitted probabilities, we sorted them and plotted them as shown in the figure 25. Please note that the x-axis is intentionally scaled as a log for observations.

As described in section 4.1, we are interested in selecting the comments with high probability. As we move towards comments with lower probability, we make the 'curriculum' harder. So, the comments with a probability of less than 0.5 are not useful. From the figure 25 we observed that the datasets, namely IWG and GermEval 2017, have a significantly lower number of comments with 'good' probability values. The other datasets have a significant number of 'useful' comments, which we can use for the next step, i.e. the Multi-task learning.

4.4 Discriminative data selection

We want to select a maximum number of comments with good probabilities from the shortlisted datasets. We call the probability value below which corresponding comments

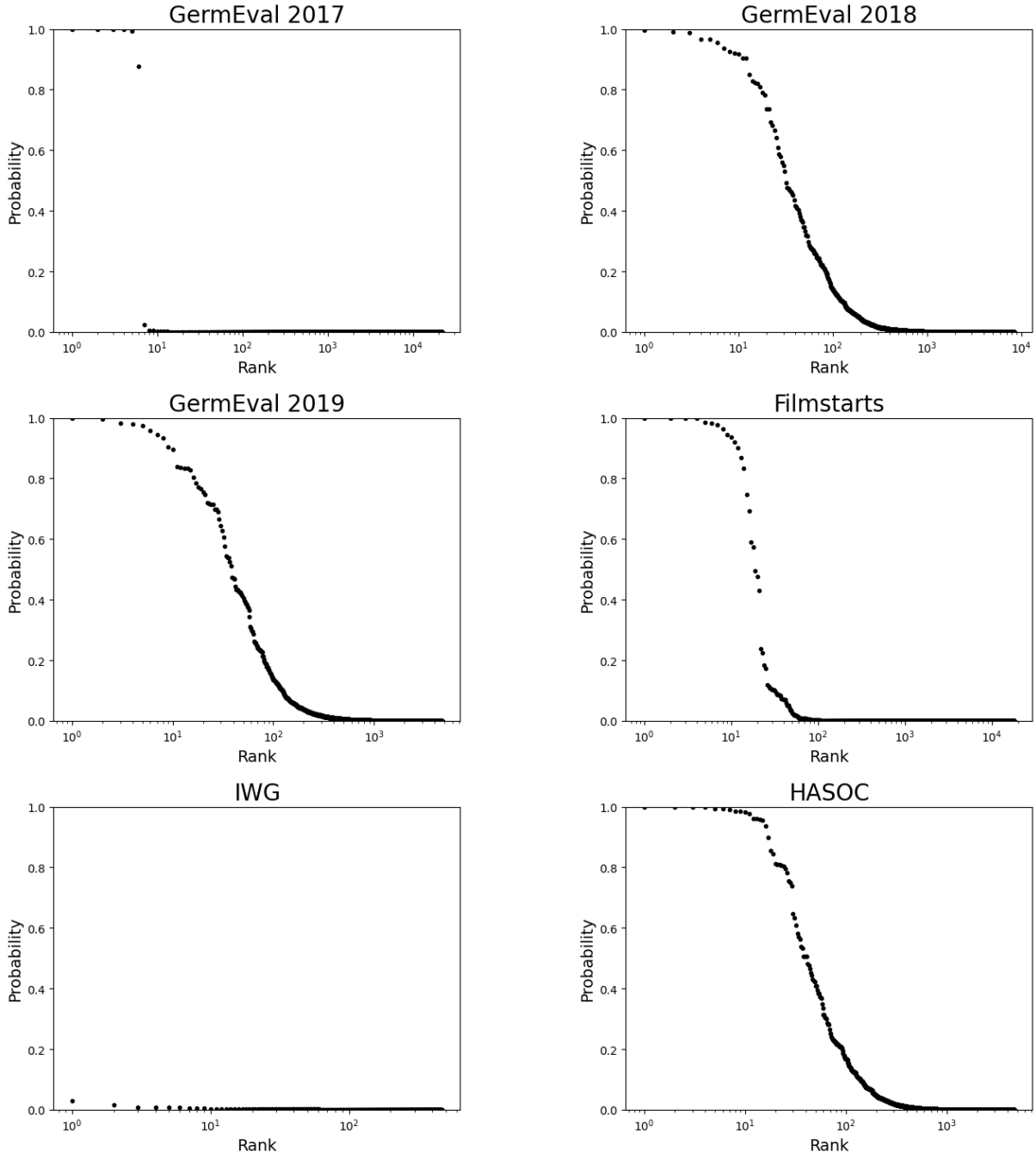


Figure 25: Plots are showing probabilities given by the domain classifier for all the short-listed datasets. The X-axis shows the rank of the comment, and the Y-axis shows the probability of the comment belonging to the target dataset. The GermEval 2017 and IWG datasets have a very few data points with more than 0.5 probability value. So, we discarded them in the disciminative data selection step.

are pruned as the 'cutoff probability'. We decided to keep the cutoff probability value to be the same across all the datasets. Taking a cutoff probability value closer to 1 will reduce the number of comments available for training. That will result in an over-fitting issue. On the other hand, increasing the cutoff probability will result in more comments available for training. However, that will result in a poor 'curriculum'. So, there is a trade-off between the cutoff probability and the training data size.

We evaluated the performance of the classifier for different values of the cutoff probability. Figure **??** summarizes the results. We observed that the performance is optimal when the cutoff probability is 0.66. So, we decided to fix it to that value for all the datasets in the remaining experiment. The 'cutoff probability' needs to be more than 0.5 due to the overconfident nature of the neural networks [Guo et al. (2017)]. Since the dataset is fairly balanced, we think it is appropriate to measure the performance in terms of accuracy.

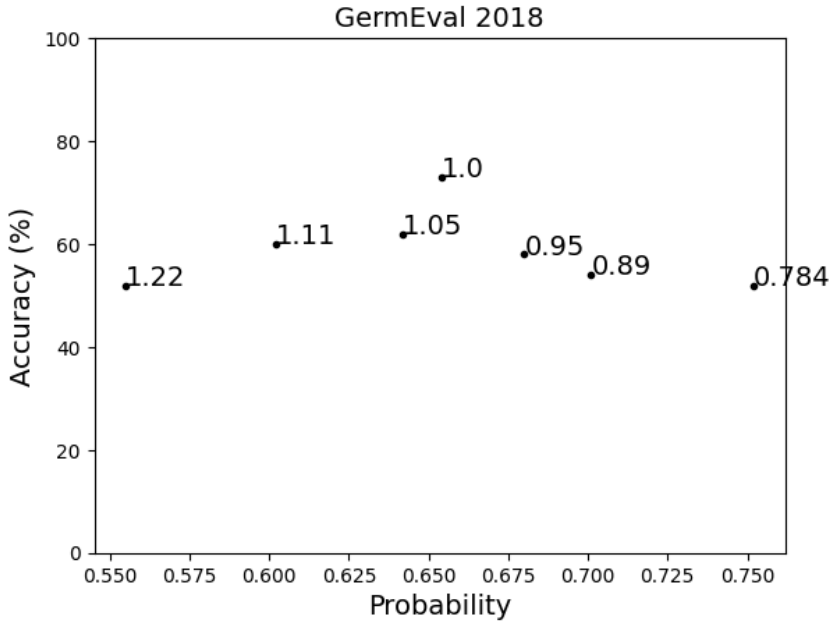


Figure 26: Result of the experiment done for deciding the cutoff probability. The data point labels show the percentage of the dataset selected. Note: We had two iterations of the Eternio dataset. The cut-off probability experiment was done for the old Eternio data.

The table 8 gives details about the percentage we selected for the next step. The figure **??** shows t-SNE visualization for the selected datasets.

Dataset	Selected data (%)
HASOC	0.759
GermEval 2019	0.412
GermEval 2018	0.292
Filmstarts	0.091
GermEval 2017	0.000
IWG	0.000

Table 8: The sorted list of datasets according to the percentage of the selected data. A very small fraction of data is selected indicating high domain shift with respect to the Eternio dataset. Note: We had two iterations of the Eternio dataset. The cut-off probability experiment (figure **??**) was done for the old Eternio data. This table shares numbers for the new Eternio dataset.

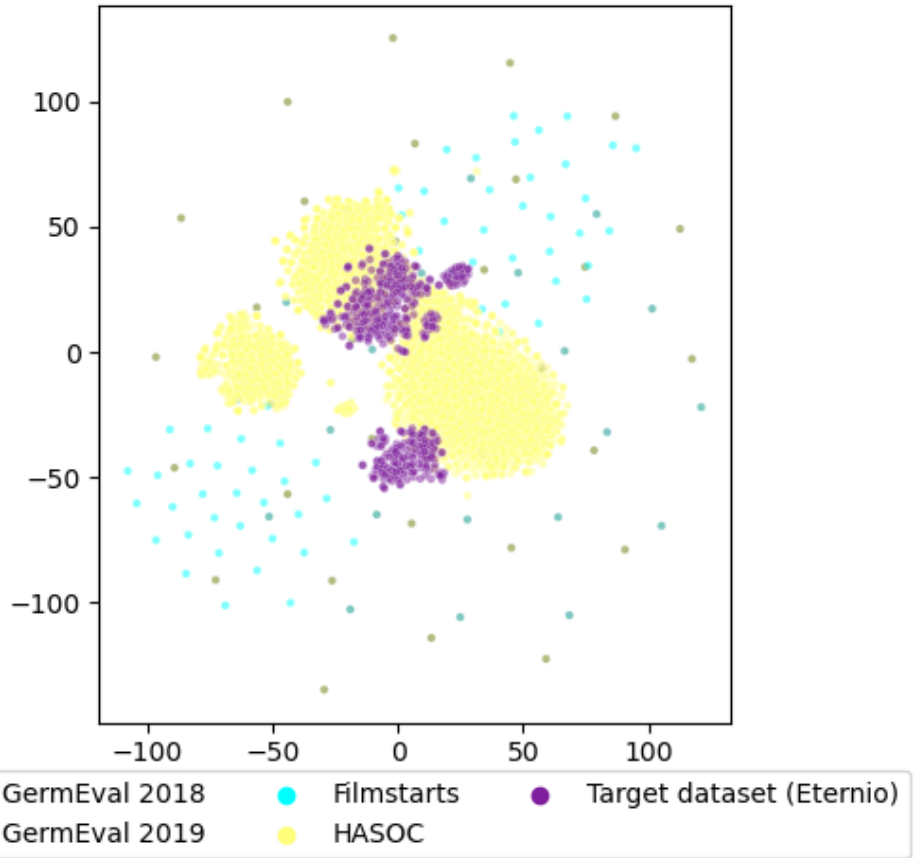


Figure 27: t-SNE visualization of the selected data points from the datasets in table 8 (perplexity=30). This plot verifies the result of the classification step as the selected data points are close the Eternio data points.

4.5 Multi-task learning

Using the selected datasets in the previous step, we designed a multi-task learning setup as shown in the figure 28. It had the pre-trained German BERT as a shared component and four task-specific layers corresponding to each dataset. We used HuggingFace's BertTokenizer for creating the tokenized input.

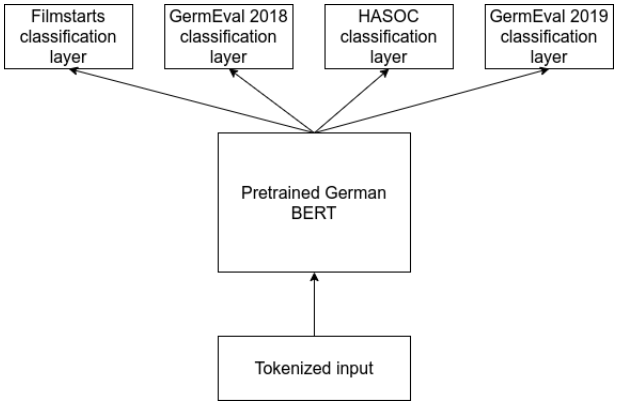


Figure 28: Our multi-task learning setup takes in the tokenized input into the pre-trained BERT. The hidden states from the final layer of the BERT are used to train the task specific classification layers.

4.6 Result and analysis

We conducted an extensive hyper-parameter search for all combinations of the shortlisted datasets. The figure 29 shows a summary of performances across all those combinations. The best performance in terms of accuracy was achieved for the combination of HASOC, GermEval 2018 and Filmstarts. The detailed classification report is as given in the table 9. The AUROC value was 0.95.

	Precision	Recall	F1-score	Support
non-offensive	0.81	0.95	0.88	212
offensive	0.95	0.83	0.89	268
accuracy			0.88	480
macro avg	0.88	0.89	0.88	480
weighted avg	0.89	0.88	0.88	480

Table 9: The Sci-kit classification report for the combination of HASOC, GermEval 2018 and Filmstarts dataset. The high precision for the offensive comments and high recall for non-offensive comments indicate good performance.

The classification report gives clues about how the model can be improved further. As given in the report, the precision for the non-offensive comments is 0.81, and the recall for the offensive comments is 0.83. Our model is falling short of detecting some of the offensive comments. We manually went through all the misclassified examples and analyzed

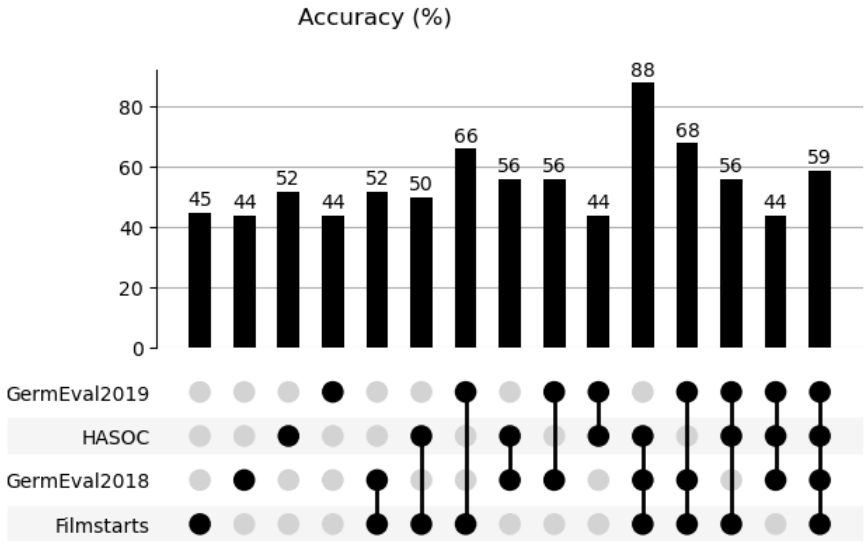


Figure 29: Performance summary for the various combination of tasks in the multi-task learning setup. We obtained the highest performance with the combination of HASOC, GermEval 2018 and Filmstarts dataset.

them. The table 10 shows types of comments where the model fails. We can improve the performance by expanding the training dataset with those kinds of examples.

Category	**Example(s)**
Irony	Bei jedem Buch das er gelesen hat, habe ich immer gehofft, das es sein letztes ist. Das es dann ausgerechnet: SStirb langsam - das Buch zum Film" sein musste, ...
	Da hätte ich mir von den Kosten für seinen Rückflug mal besser neue Schuhe gekauft. Und die Jetski-Runde hätten wir am ersten Tag machen können. Aber ab Tag drei wars so ein super Urlaub!
Sarcasm	Jetzt hat der Tod ihn an der Backe Und wir sind ihn zum Glück los
	Du bist ein schrooer Lummel
Missing context	Angela Merkel hätte das gleiche verdient
Famous quote	Die Toten sind nicht fort sie gehen mit Unsichtbar sind sie nur unhörbar ist ihr Schritt Ihr habt jetzt Trauer aber ich werde Euch wiedersehen und Euer Herz wird sich freuen Johannes 16 22

Table 10: Category of examples where our model fails.

5 Approach and analysis for the hateful meme detection

We extended the text-only Eternio offensive content detection problem to the hateful meme detection problem. This section explains various approaches tried by us and the conclusion that we drew from our experiments.

5.1 Details about the dataset

Before we proceeded with trying out new approaches, it was necessary to collect details about the dataset. It was necessary to check that the dataset does not have shortcomings like the previous ones, as discussed in section 3.4. This section gives relevant important information about the dataset from this point of view.

First, we would like to give details about the composition of the dataset. Kiela et al. (2020) classified memes into five categories: multimodal hate, unimodal hate, benign image, benign text confounders and non-hateful examples. The multimodal hateful memes have corresponding confounders for both modalities. The total size of the dataset is 10,000, with 5% (500) examples make up the dev set and 10% (1000) from the test set. See table 11 for more details.

	Total	**Non-hate**	**Hate**	**MM Hate**	**UM Hate**	**Img Conf**	**Txt Conf**	**Rand Benign**
Train	8500	5481	3019	1100	1919	1530	1530	2421
Dev	540	340	200	200	0	170	170	0
Test	2000	1250	750	750	0	625	625	0

Table 11: Composition of the hateful meme dataset [Kiela et al. (2021)].

Next, we check the presence of priors which was done by plotting t-SNE visualizations. As discussed in section 3, priors result in biased models. Das et al. (2020) did the t-SNE visualization of the language and vision modality as shown in the figure 30. In the case of vision modality, we obtained the image encoding using ResNet-152 convolution features. As one can see from the plot, it is difficult to separate hateful (dark red) and non-hateful (yellow) meme images. For language modality, sentence-level encoding was done using BERT. In this case, the separation between labels is better. However, the model performance was still not satisfactory with accuracy of 59.2% [Das et al. (2020)]. The plots in figure 30 were created with a perplexity value of 5 for the visual modality and 50 for the text modality.

As one can see in the figures 30 and 31, the hateful and non-hateful examples overlap each other. Hence, the models depending on the priors, will fail in this case. These results confirm the claim made by Facebook.

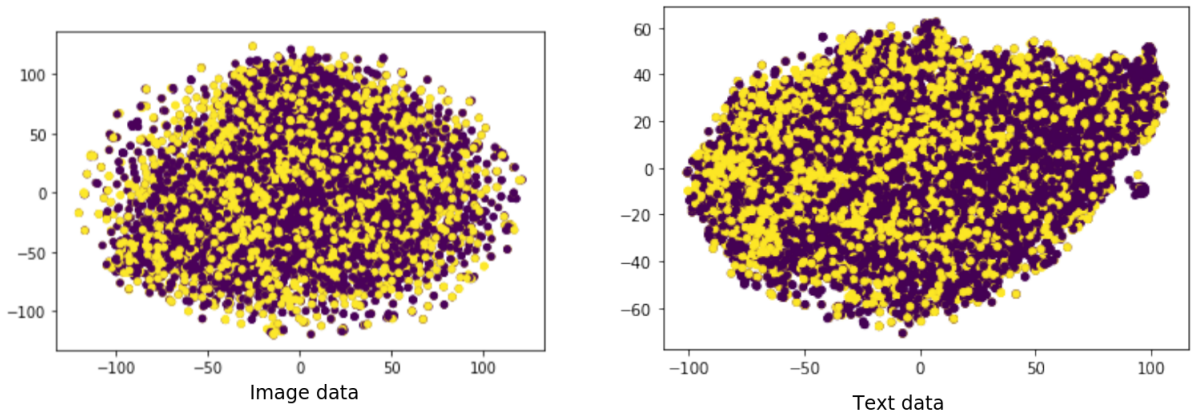


Figure 30: t-SNE visualization of the vision and language modality (sentence level) of the hateful meme dataset [Das et al. (2020)]. The hateful (dark red) and non-hateful (yellow) labels do not form separate clusters implying absence of priors.

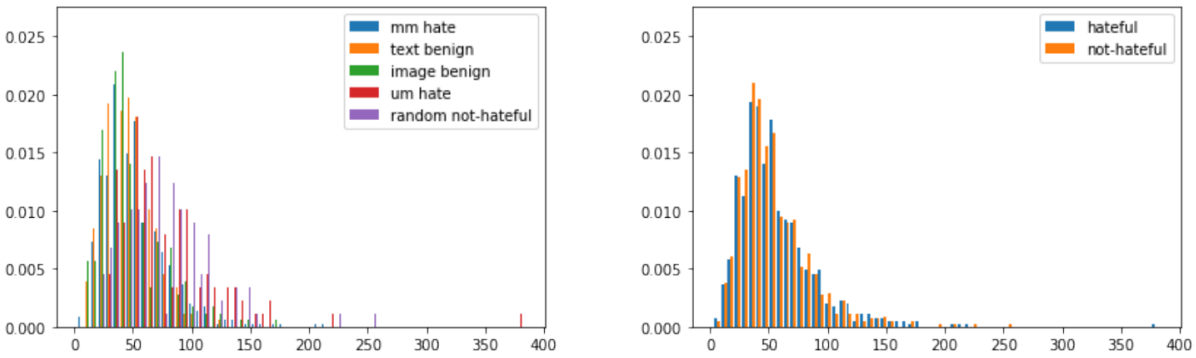


Figure 31: The normalized token count distribution for the hateful meme dataset [Kiela et al. (2020)]. It confirms absence of text based priors.

In order to develop a generic and close to the real-world model, the dataset needs to have examples from diverse categories. The dataset satisfies this requirement with the inclusion of various types of memes. Table 12 shows distribution of the memes. Note that the sum of percentages goes beyond 100 as some memes fall in more than one category. The dataset is not exhaustive but seems diverse enough.

Kiela et al. (2020) collected textual and visual lexical statistics for the dataset. The figure 31 shows token count distribution i.e. a word-level statistics. It shows that word count should not have an effect as a prior. Please refer Kiela et al. (2020) for more statistics.

5.2 Reproduction and analysis of Zhu's results

Zhu ensembled VL-BERT, UNITER-ITM, VILLA-ITM and ERNIE-Vil [Zhu (2020)]. As per the hateful meme challenge scoreboard, the approach scored 0.8450 AUROC and 73.20% accuracy. We tried to reproduce his results and analyze them. This subsection

Hate speech type	%
Comparison to animal	4.0
Comparison to object	9.2
Comparison with criminals	17.2
Exclusion	4.0
Expressing Disgust/Contempt	6.8
Mental/physical inferiority	7.2
Mocking disability	6.0
Mocking hate crime	14.0
Negative stereotypes	15.6
Other	4.4
Use of slur	2.0
Violent speech	9.6

Table 12: Hate speech categories. Note than the sum of percentages goes beyond 100 as some memes fall in more than one category [Kiela et al. (2020)].

provides details about that.

The Sci-kit classification report for the reproduced results was as given in the table 13. We achieved an AUROC score of 0.6816. Our results fell short of the results posted by Zhu. One possible explanation could be that a limited number of GPUs were available when running the experiment.

	Precision	Recall	F1-score	Support
Non-hateful	0.74	0.89	0.81	1250
Hateful	0.72	0.48	0.57	750
Accuracy			0.73	2000
Macro avg	0.73	0.68	0.69	2000
Weighted avg	0.73	0.73	0.72	2000

Table 13: Classification report for the Zhu's approach. The reproduced model gave poor numbers compared to the original model.

Next, we tried to analyze the results. We went through all misclassified memes and observed that we could improve the model by incorporating the context better. We cannot paste the exact meme here as they are extremely mean, but we will explain our point using the figure 11. In the figure 11, our model needs the context associated with a flower and smell to identify it has a hateful meme.

The remaining section discusses how we tried out different ideas for capturing the context better.

5.3 Approaches

As discussed in section 3, the hateful meme detection problem is similar to the VQA problem. So, we tried to try out some of the top-performing VQA problems. This subsection discusses the results.

5.3.1 MUTAN

Since we want to incorporate the outside knowledge and context, the best performing approaches on the OK-VQA (Outside Knowledge VQA) dataset seemed helpful. As per the leaderboard, the MUTAN fusion [Ben-Younes et al. (2017)] gives a good performance. Since the top approach, i.e. ConceptBERT [Garderes et al. (2020)], did not provide the source code, we decided to go with MUTAN.

We have already discussed MUTAN fusion in the section 3.7.2. We decided to encode the questions using BERT, and image features were extracted using ResNet-152. The model performed poorly, as evident from the table 14. It achieved an AUROC of 0.60.

	Precision	Recall	F1-score	Support
Non-hateful	0.69	0.83	0.75	1250
Hateful	0.56	0.38	0.45	750
Accuracy			0.66	2000
Macro avg	0.63	0.60	0.60	2000
Weighted avg	0.64	0.66	0.64	2000

Table 14: The Sci-kit classification report for the MUTAN shows that it performed poorly. The poor recall number for the hateful meme class imply its failure to identify hateful memes.

The performance numbers are as good as a random classifier. We analyzed all the misclassified examples manually by going through them one by one, but we did not observe any pattern. We also plotted the t-SNE plot (figure 32), which did not reveal any clustering pattern.

5.3.2 ReGAT

We have discussed ReGAT in section 3.7.2. Its ability to capture implicit relations between visual objects might help to capture context better. So, we conducted experiments with implicit features, and it performed as given in the table 15. We could not conduct experiments with semantic and spatial feature extractors because the source code was not available.

The ReGAT model achieved an AUROC of 0.6329, and the best performance was achieved with BAN fusion. We analyzed the misclassified memes manually, but we did not observe

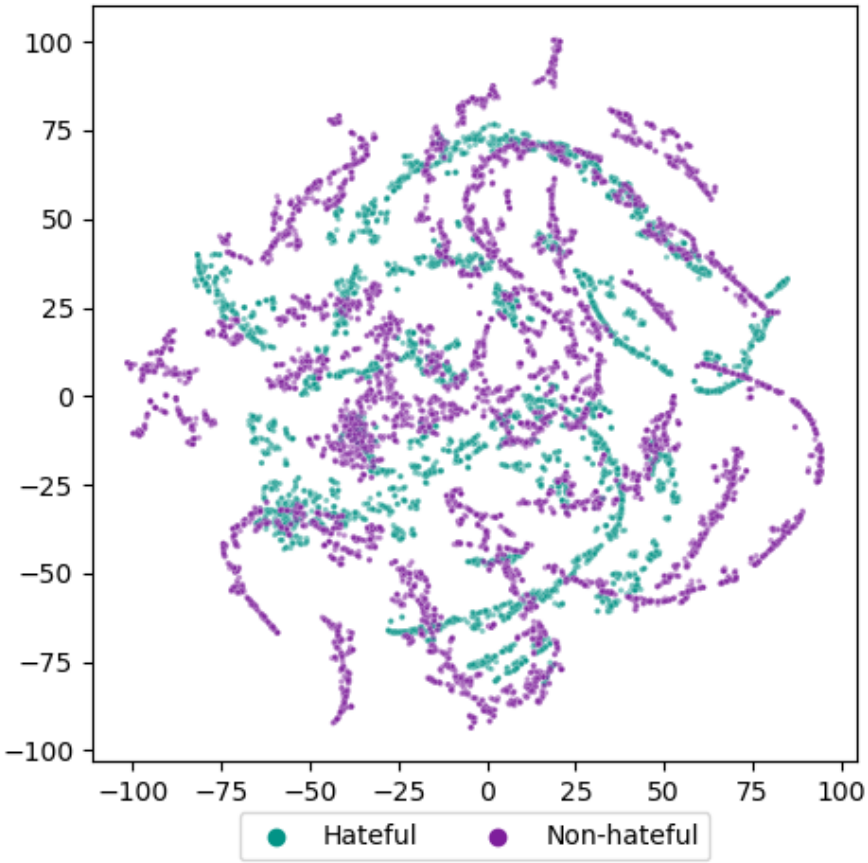


Figure 32: The t-SNE plot of the multi-modal embedding before the classification layer did not show any clustering effect (perplexity = 20).

	Precision	Recall	F1-score	Support
Non-hateful	0.74	0.66	0.70	1250
Hateful	0.52	0.61	0.56	750
accuracy			0.64	2000
macro avg	0.63	0.63	0.63	2000
weighted avg	0.65	0.64	0.64	2000

Table 15: The Sci-kit classification report for ReGAT with implicit features performed poorly. The poor recall number on hateful meme implies its inability to identify hateful memes.

any pattern to conclude confidently. The t-SNE plot also did not show any clustering (figure 33).

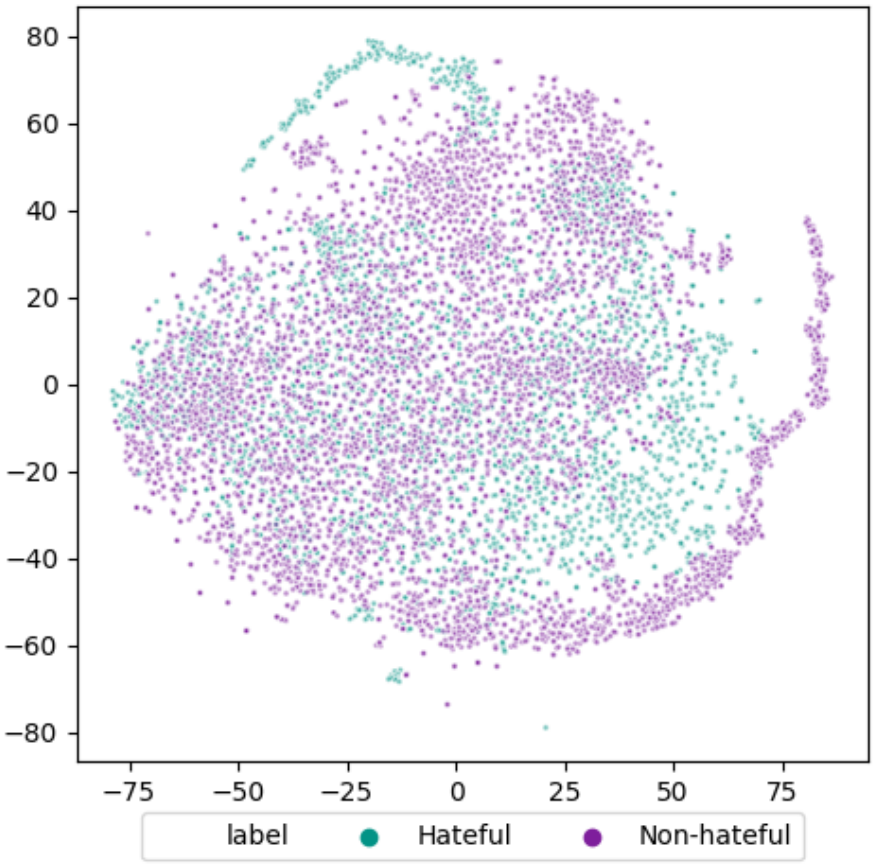


Figure 33: The t-SNE plot using the embedding in the latent space before the classification layer (perplexity = 20).

5.4 Future work

The hateful meme detection problem was part of the NeurIPS competition track for the year 2020. This competition has opened up many possibilities for future work. The first possible direction of future work is improving the identification of the hateful meme. We have discussed the problem from the perspective of improving the performance in terms of AUROC. As we discussed in the previous sections, the MUTAN and ReGAT approaches performed poorly. The approaches proposed by winners of the hateful meme challenge have performed good, but the challenge is still far from being solved as the approaches are far behind the human performance in terms of accuracy (84.7%). The advanced fusion techniques also fall short of giving improvement [Yang et al. (2019)]. Overall, we will need advanced pre-trained models to understand better' context to solve this challenge.

To solve the challenge, Muennighoff (2020) built a framework from scratch. Zhu (2020) ported PaddlePaddle model to PyTorch. This highlights technical challenges in solving

the multimodal classification problem. Facebook has been developing the MMF (MultiModal Framework) framework and the hateful meme dataset to solve this problem. The availability of a good framework should accelerate the vision and language domain research.

The various approaches proposed by the participants use pre-trained models. The participants that use more recent pre-trained models have scored better scores on the leaderboard. They attribute this is advancement in architecture but do not provide strong evidence for it. Investigating what the model has learned can give us a better idea of why the winning idea worked and what can be done further to improve it.

Overall, the hateful meme dataset has opened up many possibilities for future work.

6 Conclusion

This thesis work was conducted in two parts. In the first part, we developed a text-only offensive comment detector for Eternio GmbH. In the second part, we tried to extend the problem by including visual modality in the comments. We discuss conclusion for the each of the parts one by one.

In the first part, we implemented the domain adaptation with the data selection approach published by Ma et al. (2019). We successfully extended the approach with Multi-task learning. The performance of the model was sufficient but can be improved further by including the sarcasm dataset. The approach worked because the BERT has a good domain discriminating nature, helping the classification task. As a result, the data selection was accurate. One important observation during the data selection process is that the cutoff probability needs to be higher than 0.5. This can be attributed to the overconfident nature of the neural networks. Next, in the MTL step, the knowledge of language encoded in BERT [Rogers et al. (2021)] helped in achieving good performance. Finally, we hope this work helps software developers from German startup companies solve offensive comment detection problems in their products.

In the second part, we considered the case of hateful memes (in English) published by Facebook researchers. We surveyed various approaches tried by the Hateful meme challenge participants and tried a few VQA models. The ensemble approach is the current best performing approach, but to improve further, we need advanced architecture. Overall, we agree with the Kiela et al. (2021) that multimodal hate speech remains a challenging problem that demands new approaches and methods for solving.

References

Agrawal, A., Batra, D., Parikh, D., and Kembhavi, A. (2018). Don't just assume; look and answer: Overcoming priors for visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4971–4980.

Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Zitnick, C. L., and Parikh, D. (2015). Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433.

Baltrušaitis, T., Ahuja, C., and Morency, L.-P. (2018). Multimodal machine learning: A survey and taxonomy. *IEEE transactions on pattern analysis and machine intelligence*, 41(2):423–443.

Ben-Younes, H., Cadene, R., Cord, M., and Thome, N. (2017). Mutan: Multimodal tucker fusion for visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2612–2620.

Caruana, R. (1997). Multitask learning. *Machine learning*, 28(1):41–75.

Chen, Y.-C., Li, L., Yu, L., Kholy, A. E., Ahmed, F., Gan, Z., Cheng, Y., and Liu, J. (2020). Uniter: Universal image-text representation learning.

Csurka, G. (2017). A comprehensive survey on domain adaptation for visual applications. *Domain adaptation in computer vision applications*, pages 1–35.

Das, A., Kottur, S., Gupta, K., Singh, A., Yadav, D., Moura, J. M., Parikh, D., and Batra, D. (2017). Visual Dialog. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Das, A., Wahi, J. S., and Li, S. (2020). Detecting hate speech in multi-modal memes. *arXiv preprint arXiv:2012.14891*.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT (1)*.

Gan, Z. (2020). Visual qa and reasoning. Available at https://rohit497.github.io/Recent-Advances-in-Vision-and-Language-Research/slides/tutorial-part-2-vqa.pdf (2020/06/15).

Gan, Z., Chen, Y.-C., Li, L., Zhu, C., Cheng, Y., and Liu, J. (2020). Large-scale adversarial training for vision-and-language representation learning. *arXiv preprint arXiv:2006.06195*.

Garderes, F., Ziaeefard, M., Abeloos, B., and Lecue, F. (2020). Conceptbert: Concept-aware representation for visual question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 489–498.

Gomez, R., Gibert, J., Gomez, L., and Karatzas, D. (2020). Exploring hate speech detection in multimodal publications. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1470–1478.

Goyal, Y., Khot, T., Summers-Stay, D., Batra, D., and Parikh, D. (2017). Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6904–6913.

Guhr, O., Schumann, A.-K., Bahrmann, F., and Böhme, H. J. (2020). Training a broad-coverage german sentiment classification model for dialog systems. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 1627–1632.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. (2017). On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1321–1330.

Gurari, D., Li, Q., Stangl, A. J., Guo, A., Lin, C., Grauman, K., Luo, J., and Bigham, J. P. (2018). Vizwiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3608–3617.

Hinton, G. and Roweis, S. T. (2002). Stochastic neighbor embedding. In *NIPS*, volume 15, pages 833–840. Citeseer.

Hudson, D. A. and Manning, C. D. (2019). Gqa: A new dataset for real-world visual reasoning and compositional question answering. *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Jiang, Y., Natarajan, V., Chen, X., Rohrbach, M., Batra, D., and Parikh, D. (2018). Pythia v0. 1: the winning entry to the vqa challenge 2018. *arXiv preprint arXiv:1807.09956*.

Jigsaw, C. A. (2018). Toxic Comment Classification Challenge.

Jigsaw, C. A. (2019). Jigsaw Unintended Bias in Toxicity Classification.

Khodak, M., Saunshi, N., and Vodrahalli, K. (2017). A large self-annotated corpus for sarcasm. *arXiv preprint arXiv:1704.05579*.

Kiela, D., Firooz, H., Mohan, A., Goswami, V., Singh, A., Fitzpatrick, C. A., Bull, P., Lipstein, G., Nelli, T., Zhu, R., et al. (2021). The hateful memes challenge: Competition report. In *NeurIPS 2020 Competition and Demonstration Track*, pages 344–360. PMLR.

Kiela, D., Firooz, H., Mohan, A., Goswami, V., Singh, A., Ringshia, P., and Testuggine, D. (2020). The hateful memes challenge: Detecting hate speech in multimodal memes. *arXiv preprint arXiv:2005.04790*.

Kobayashi, S. (2018). Contextual augmentation: Data augmentation by words with paradigmatic relations. *arXiv preprint arXiv:1805.06201*.

Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.-J., Shamma, D. A., et al. (2017). Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73.

Li, L., Gan, Z., Cheng, Y., and Liu, J. (2019). Relation-aware graph attention network for visual question answering. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 10312–10321. IEEE.

Li, X., Yin, X., Li, C., Zhang, P., Hu, X., Zhang, L., Wang, L., Hu, H., Dong, L., Wei, F., et al. (2020). Oscar: Object-semantics aligned pre-training for vision-language tasks. In *European Conference on Computer Vision*, pages 121–137. Springer.

Liu, X., He, P., Chen, W., and Gao, J. (2019). Multi-task deep neural networks for natural language understanding. *arXiv preprint arXiv:1901.11504*.

Ma, X., Xu, P., Wang, Z., Nallapati, R., and Xiang, B. (2019). Domain adaptation with bert-based domain classification and data selection. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*, pages 76–83.

Mandl, T., Modha, S., Majumder, P., Patel, D., Dave, M., Mandlia, C., and Patel, A. (2019). Overview of the hasoc track at fire 2019: Hate speech and offensive content identification in indo-european languages. In *Proceedings of the 11th forum for information retrieval evaluation*, pages 14–17.

Mao, J., Huang, J., Toshev, A., Camburu, O., Yuille, A. L., and Murphy, K. (2016). Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 11–20.

Marino, K., Rastegari, M., Farhadi, A., and Mottaghi, R. (2019). Ok-vqa: A visual question answering benchmark requiring external knowledge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3195–3204.

Mogadala, A., Kalimuthu, M., and Klakow, D. (2021). Trends in integration of vision and language research: A survey of tasks, datasets, and methods. *Journal of Artificial Intelligence Research*, 71:1183–1317.

Muennighoff, N. (2020). Vilio: State-of-the-art visio-linguistic models applied to hateful memes. *arXiv preprint arXiv:2012.07788*.

Müller, R., Kornblith, S., and Hinton, G. E. (2019). When does label smoothing help? In *NeurIPS*.

Pan, S. J., Tsang, I. W., Kwok, J. T., and Yang, Q. (2010). Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2):199–210.

Pan, S. J. and Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359.

Pfeiffer, J., Rücklé, A., Poth, C., Kamath, A., Vulić, I., Ruder, S., Cho, K., and Gurevych, I. (2020). Adapterhub: A framework for adapting transformers. *arXiv preprint arXiv:2007.07779*.

Rogers, A., Kovaleva, O., and Rumshisky, A. (2021). A Primer in BERTology: What We Know About How BERT Works. *Transactions of the Association for Computational Linguistics*, 8:842–866.

Rosen, G. (2020). Community standards enforcement report, november 2020.

Ross, B., Rist, M., Carbonell, G., Cabrera, B., Kurowsky, N., and Wojatzki, M. (2017). Measuring the reliability of hate speech annotations: The case of the european refugee crisis. *arXiv preprint arXiv:1701.08118*.

Ruder, S. (2017). An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*.

Sabat, B. O., Ferrer, C. C., and Giro-i Nieto, X. (2019). Hate speech in pixels: Detection of offensive memes towards automatic moderation. *arXiv preprint arXiv:1910.02334*.

Schmidt, A. and Wiegand, M. (2017). A survey on hate speech detection using natural language processing. In *Proceedings of the fifth international workshop on natural language processing for social media*, pages 1–10.

Shah, M., Chen, X., Rohrbach, M., and Parikh, D. (2019). Cycle-consistency for robust visual question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6649–6658.

Sharma, C., Bhageria, D., Scott, W., PYKL, S., Das, A., Chakraborty, T., Pulabaigari, V., and Gamback, B. (2020). Semeval-2020 task 8: Memotion analysis–the visuo-lingual metaphor! *arXiv preprint arXiv:2008.03781*.

Singh, A., Natarajan, V., Shah, M., Jiang, Y., Chen, X., Batra, D., Parikh, D., and Rohrbach, M. (2019). Towards vqa models that can read. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8317–8326.

Struß, J. M., Siegel, M., Ruppenhofer, J., Wiegand, M., Klenner, M., et al. (2019). Overview of germeval task 2, 2019 shared task on the identification of offensive language. -.

Su, W., Zhu, X., Cao, Y., Li, B., Lu, L., Wei, F., and Dai, J. (2019). Vl-bert: Pre-training of generic visual-linguistic representations. *arXiv preprint arXiv:1908.08530*.

Suhr, A., Zhou, S., Zhang, A., Zhang, I., Bai, H., and Artzi, Y. (2019). A corpus for reasoning about natural language grounded in photographs. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6418–6428, Florence, Italy. Association for Computational Linguistics.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826.

Tan, H. and Bansal, M. (2019). Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*.

Van der Maaten, L. (2013). Visualizing data using t-sne.

Van der Maaten, L. and Hinton, G. (2008). Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

Velioglu, R. and Rose, J. (2020). Detecting hate speech in memes using multimodal deep learning approaches: Prize-winning solution to hateful memes challenge. *arXiv preprint arXiv:2012.12975*.

Vijayaraghavan, P., Larochelle, H., and Roy, D. (2021). Interpretable multi-modal hate speech detection. *arXiv preprint arXiv:2103.01616*.

Wang, Y., Yao, Q., Kwok, J. T., and Ni, L. M. (2020). Generalizing from a few examples: A survey on few-shot learning. *ACM Computing Surveys (CSUR)*, 53(3):1–34.

Wattenberg, M., Viégas, F., and Johnson, I. (2016). How to use t-sne effectively. *Distill*.

Wiegand, M. (2019). GermEval-2018 Corpus (DE).

Wojatzki, M., Ruppert, E., Holschneider, S., Zesch, T., and Biemann, C. (2017). Germeval 2017: Shared task on aspect-based sentiment in social media customer feedback. *Proceedings of the GermEval*, pages 1–12.

Wu, Y., Kirillov, A., Massa, F., Lo, W.-Y., and Girshick, R. (2019). Detectron2. `https://github.com/facebookresearch/detectron2`.

Xie, N., Lai, F., Doran, D., and Kadav, A. (2019). Visual entailment: A novel task for fine-grained image understanding.

Yan, L., Dodier, R. H., Mozer, M., and Wolniewicz, R. H. (2003). Optimizing classifier performance via an approximation to the wilcoxon-mann-whitney statistic. In *Proceedings of the 20th international conference on machine learning (icml-03)*, pages 848–855.

Yang, F., Peng, X., Ghosh, G., Shilon, R., Ma, H., Moore, E., and Predovic, G. (2019). Exploring deep multimodal fusion of text and photo for hate speech classification. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 11–18.

Yu, F., Tang, J., Yin, W., Sun, Y., Tian, H., Wu, H., and Wang, H. (2020). Ernie-vil: Knowledge enhanced vision-language representations through scene graph. *arXiv preprint arXiv:2006.16934*.

Zhang, Y. and Yang, Q. (2021). A survey on multi-task learning. *IEEE Transactions on Knowledge and Data Engineering*.

Zhu, R. (2020). Enhance multimodal transformer with external label and in-domain pretrain: Hateful meme challenge winning solution. *arXiv preprint arXiv:2012.08290*.