Memebusters at SemEval-2020 Task 8: Feature Fusion Model for Sentiment Analysis on Memes using Transfer Learning

Mayukh Sharma, Ilanthenral Kandasamy, W.B. Vasantha
School of Computer Science and Engineering
Vellore Institute of Technology
Vellore, Tamil Nadu, India
mayukh.sharma2016@vitstudent.ac.in,
ilanthenral.k@vit.ac.in, vasantha.wb@vit.ac.in

Abstract

In this paper, we describe our deep learning system used for SemEval 2020 Task 8: Memotion analysis. We participated in all the subtasks i.e Subtask A: Sentiment classification, Subtask B: Humor classification, and Subtask C: Scales of semantic classes. Similar multimodal architecture was used for each subtask. The proposed architecture makes use of transfer learning for images and text feature extraction. The extracted features are then fused together using stacked bi-directional Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU) model with attention mechanism for final predictions. We also propose a single model for predicting semantic classes (Subtask B) as well as their scales (Subtask C) by branching the final output of the post LSTM dense layers. Our model was ranked 5 in Subtask B and ranked 8 in Subtask C and performed nicely in Subtask A on the leader board. Our system makes use of transfer learning for feature extraction and fusion of image and text features for predictions.

1 Introduction

A meme is an approach, concept, idea or style that spreads through social media within a society often with the goal of expressing a trend, topic, or significance represented by the meme (Peirson and Tolunay, 2018). Recent developments in social media analytics have shown a widespread increase in the usage of memes in expressing sentiment. Social media networks like Facebook, Instagram and Twitter usually get flooded with memes upon occurrence of any popular event. Memes are present in almost every form of media and they are constantly evolving considering the events happening around the world. The spread of memes on social media can be considered analogous to human genetics. Slightly altered versions of the same memes spread in social media in a similar way as copies of human genes pass through human lineage (French, 2017). They work as a medium for sharing humour on the pretext of cultural themes. They represent the sentiment of a community or culture with respect to the event involved. This is beneficial in gauging the sentiment of people. Memes concerning to political events may represent the political outlook of a community. Simultaneously, they can also be engineered to further political ideals, amplify echo chambers and alienate minorities. They are altogether a different means of communication and have become fundamental part of the current generation.

Memes are uniquely multimodal, they contain images in conjugation with textual comments and conversations to add emphasis and provide additional meaning. Rarely, memes contain only the text or visual component. Thus, they pose strong challenge in identifying their semantics which may involve use of text as well as image features. Significant research has been carried out on sentiment classification of twitter data (Rosenthal et al., 2017), but they mostly confine to only the text. In comparison, multimodal analysis of social media content has very little research on it. Meme text generation (Peirson and Tolunay, 2018) on image templates provides some useful insights on use of deep learning on memes. The SemEval 2020 Task 8: Memotion analysis (Sharma et al., 2020) draws attention towards analyzing sentiment of memes on social media to extract the conveyed message and identify the semantics of the memes.

This work is licensed under a Creative Commons Attribution 4.0 International Licence. Licence details: http://creativecommons.org/licenses/by/4.0/.

Proceedings of the 14th International Workshop on Semantic Evaluation, pages 1163–1171
Barcelona, Spain (Online), December 12, 2020.

In this paper we describe our deep learning model that competed in the SemEval 2020 Task 8: Memotion analysis (Sharma et al., 2020). Our proposed system makes use of transfer learning as a key component for extracting features from meme images and their Optical Character Recognition (OCR) extracted text. We use stacked bi-directional Long Short Term Memory (Bi-LSTM), GRU and attention mechanism for fusing the features of the two modalities into a single feature space. We use these combined features for classifying the humour class as well as quantify them on the given semantic scale.

Our model performed well for Subtask B and Subtask C. We were ranked 5^{th} and 8^{th} on the official leader board. We propose a single multitask learning model for joint feature extraction which can then be used for multiple tasks dependent on similar feature space. This removes the need for training different systems for Subtask B and Subtask C. This brings down computational cost and complexity. Moreover, it advances the idea of transfer learning for tasks dependent on similar semantics. Our model was ranked 23^{th} in Subtask A. One possible reason for comparatively low performance could be the inability to differentiate between neutral and positive classes as they maybe remarkably similar.

Our code is available online [1] for method replicability.

2 Background

Understanding semantics of memes requires multimodal analysis, since they contain an image with some text. Occasionally it is an image without text that is symbolic of an emotion, such as surprise or joy (French, 2017), it poses an important challenge in recognizing their semantics. We need to take both components of a meme to analyse them. The study done in (French, 2017) focused on classifying the relationship between meme text and its corresponding image from social media. Sentiment classification using images combined with their label embeddings (Graesser et al., 2017) outperformed both only image and only text models.

The text only model's performance was comparable with combined (image and text) model, which implies that textual data plays a crucial role while analysing multimodal data. Combining of the text and image modalities also poses a major challenge while performing analysis on meme data. Another challenge in performing analysis on such data is combining the two different modalities i.e text and image. Work done by (Duong et al., 2017) involves different fusion techniques for effectively combining the data and discusses their performance. (Hu and Flaxman, 2018) applied deep learning techniques to investigate the structure of emotions in multimodal data. They used transfer learning for extracting the features of each modality and concatenated them for further classification. Another important study on memes involved generating captions for meme templates (Peirson and Tolunay, 2018). It made use of image captioning model proposed by (Vinyals et al., 2014) which uses an encoder-decoder network. Work presented at (Rosenthal et al., 2017), (Zampieri et al., 2019) provides useful insights for understanding the sentiment represented in text data.

Memes have become an inherent part of modern-day world. Meagre research has been carried on memes and using combined text and images data for analysis. SemEval 2020 Task 8: Memotion analysis is an effort to bring research attention to this topic, it has three subtasks which we describe as follows: Subtask A (Sentiment Classification): Given a labelled dataset D of internet memes and their OCR extracted text, the objective of the task is to learn a classification function that can predict label L for a given meme, where $L \in \{$negative, neutral, positive$\}$. Subtask B (Humour classification): Given a labelled dataset D of internet memes and their OCR extracted text, the objective of the task is to learn a classification function that can identify the type of humour H for a given meme, where $H \in \{$sarcastic, humorous, offensive, motivational$\}$. Subtask C (Scales of semantic classes): Given a labelled dataset D of internet memes and their OCR extracted text, the objective of the task is to learn a classification function that can quantify the extent E to which a particular humour is expressed, where $E \in \{$not, slightly, mildly, very$\}$.

Analysis of training set: Training set provided consisted of 6992 meme images and their OCR extracted text. The images were annotated as per the requirements of each subtask. Table 1 shows the statistics of the training set. For Subtask A and Subtask B we can see that the training set is imbalanced with respect

[1] https://github.com/04mayukh/Memebusters-at-SemEval-2020-Task-8-Memotion-Analysis

Subtask	Dataset	Labels				
A		Very negative	negative	neutral	positive	Very positive
	Train	151	480	2201	3127	1033
	Validation	20	51	279	412	152
B		Class	Labels			
			negative	positive		
	Train	Sarcasm	1544	5448		
		Offensive	2713	4279		
		Humorous	1651	5341		
		Motivational	4525	2467		
	Validation	Sarcasm	233	681		
		Offensive	369	545		
		Humorous	210	704		
		Motivational	325	589		
C		Class	Labels			
	Train		not	slightly	mildly	Very
		Sarcasm	1544	3507	1547	394
		Offensive	2713	2592	1466	221
		Humorous	1651	2452	2238	651
	Validation		not	slightly	mildly	very
		Sarcasm	233	444	195	42
		Offensive	369	323	198	24
		Humorous	210	315	310	79

Table 1: Details of Dataset

to the negative class. It may lead our model to develop bias towards the positive class. Similarly, for Subtask C the “Very” semantic scale is poorly represented. To overcome this problem, we used class weights which we describe in the following section.

3 System Overview

3.1 High level Overview

Figure 1 represents a high-level overview of our system. We process each modality separately to extract their individual features. Images are resized as per the requirement of the inception network and the text data is cleaned. The individual features are then fused into a single feature space as described in the following subsections. The fused features are then used for classification and quantification.

3.2 Text pre-processing

The first step before feeding text data to any machine learning algorithm involves text pre-processing in order to clean the data. The extracted OCR text from the meme was cleaned and then processed. We followed the following pre-processing steps:

1. *Website and URL removal:* The text consisted of URL’s from which the memes were extracted. We eliminated them since they do not provide valuable information.

2. *Chat word conversion:* Chat words like “LOL”, “LMAO” etc. are extensively used for expressing emotions and can be helpful in recognizing the context. We converted them into their respective full forms.

3. *Emoticon conversion:* Emoticons are like emojis and are used in similar way. They are useful for increasing sentiment in text on social media. We converted emoticons to their respective meanings.

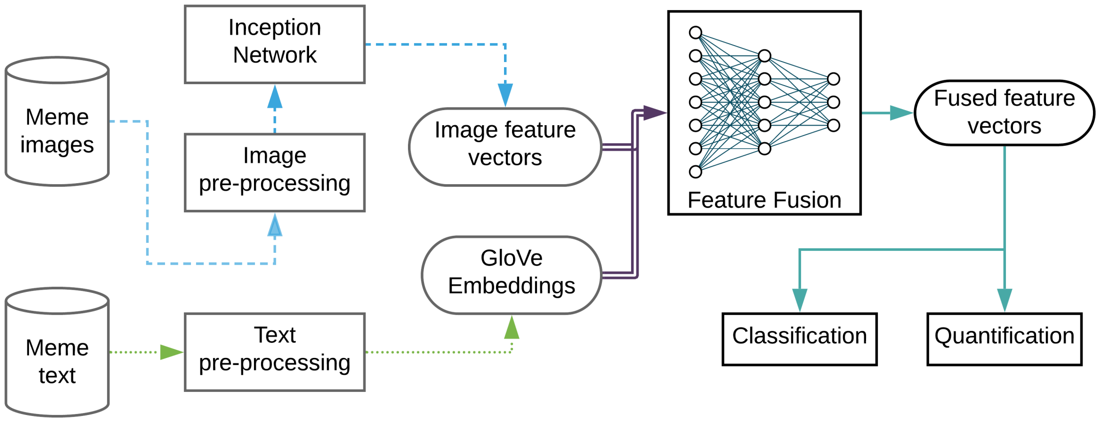

Figure 1: High level overview of our system

4. We further cleaned text data using ekphrasis (Baziotis et al., 2017) library: It normalizes the date's, time, numbers into a standard format. Annotates "hashtags", "allcaps", "elongated", "repeated", "emphasis", "censored". It performs hashtag segmentation and spelling correction on basis of twitter corpus. Lastly it tokenises the sentences.

3.3 Transfer Learning

1. *Meme images: Inception network*
   We used transfer learning for extracting features from meme images. For feature extraction from images, the pre-trained inception network (Szegedy et al., 2015) which was trained on imagenet (Russakovsky et al., 2015) dataset was used. Training a convolution network can be challenging as it is dependent on large datasets and requires testing different architectures before achieving satisfying performance. Transfer learning is a commendable approach as it decreases the computation required. The model is pre-trained on millions of images and it comprehends the fundamental composition of images.

2. *Meme text: Global Vectors for Word Representation (GloVe)*
   Word embeddings represent the semantic and syntactic meaning of the words as dense vector representations. It has improved the performance of several downstream tasks across various domains like text classification, machine comprehension etc., (Indurthi et al., 2019). GloVe (Pennington et al., 2014) embeddings based on twitter corpus was used for encoding the words of each meme text. The choice of using embeddings trained on twitter corpus owes to the fact that it contains words from movies, TV show's and important events that are used for making memes.

3.4 Recurrent neural networks [LSTM's and GRU's]

Recurrent neural networks (RNN) are effective for handling sequential information, they are called recurrent because they perform the same computation for each element, with the output being dependent on the previous computations. LSTM's (Hochreiter and Schmidhuber, 1997) and GRU (Cho et al., 2014) are frequently used RNNs. They overcome the problem of vanishing gradients which reduces the efficiency of earlier layers using several gates and cell states (Bengio et al., 1994). The cell states together with gates help in transporting relative information all the way down the sequences. The cell gates alter the sequence information. The collective effect of the memory cells along with gates aid the LSTM and GRU to learn long term dependencies.

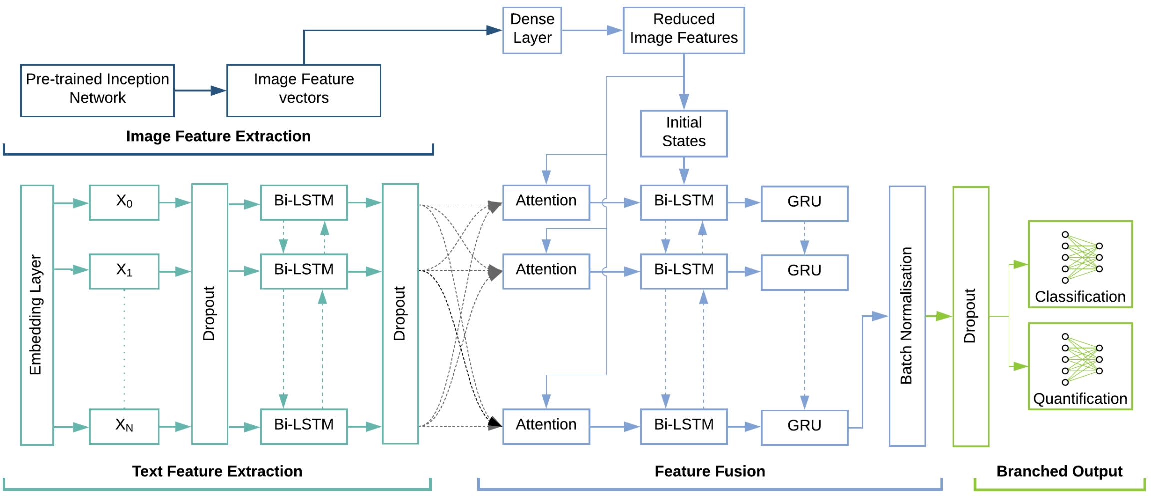

Figure 2: Architecture Diagram

3.5 Fusing image and text data

The key challenge in the task was to merge the individual features obtained from text and images. (Duong et al., 2017) recommended a technique which used pooling to bring together features of different modalities of same dimension. The feature vectors of text and images were brought to same dimension. Text features are then modelled on images using LSTM's. A comparable method for image captioning using RNN's revealed encouraging results (Vinyals et al., 2014). The image feature vectors were passed as the initial states of the LSTM. This makes the LSTM aware of the image contents. We also made use of attention (Bahdanau et al., 2014) with respect to the image feature vector for each time step of RNN to amplify the important words with respect to image.

3.6 Single model for humour classification as well as predicting it scale

Subtask B and Subtask C required us to predict humour class H as well as quantify it on a scale E. Findings of (Majumder et al., 2019) show that multitask learning-based methods significantly outperform standalone sentiment and sarcasm classifiers indicating that sentiment classification and sarcasm detection are interrelated tasks. Subtask B and Subtask C for each humour class can be considered as inter-dependent. Hence, we use a single multitask learning model.

3.7 Model Description

Our model comprises of 4 main components:

Image feature extraction: Image feature vectors were extracted using the inception network. We used the pre-final layer of the model for this purpose. Images were first resized to fit the requirements of the inception network.

Text feature extraction: We used 100-dimensional GloVe embeddings for word level vector representation of the meme text content. We then passed the embeddings through a bi-directional LSTM to obtain a richer feature representation capable of understanding the overall context of the sentence.

Feature fusion: This step involves combining the feature vectors from text and image. We use a bi-directional LSTM and attention for this task. The image feature vector dimension is reduced to the size of Bi-LSTM layer. We initialize the hidden state and cell state of Bi-LSTM with the image feature vector. Attention is calculated on text feature vectors with respect to image feature vector which is used as input for the LSTM for each timestamp. It amplifies the useful words with respect to the image and neglects the less important information. The outputs of the LSTM are then fed to a GRU. The output of the final timestamp of the GRU is then normalised using the batch normalisation layer (Ioffe and Szegedy, 2015). The output of the batch normalisation layers represents the fused feature vector.

Classification and quantification: Once the fused feature vector is obtained the next step consists of simple dense layers for prediction. The output of normalization layer is branched into two separate dense layers one for humour classification and the other for quantifying it. Subtasks for which quantification scales were unavailable consisted of a single classification layer.

Regularization: Due to the small size of the training set the model is easily prone to over-fitting. To overcome this problem, we used regularization. We used dropout (Srivastava et al., 2014) to randomly turn-off neurons in our neural network. Dropout prevents the adjacent neurons from learning similar features. It acts like ensemble learning by randomly shutting down some neurons at each iteration. For each training example only a sub-part of the network is utilized. Dropout is also used for the recurrent connections of RNN. In the final dense layer's, we used L2 regularization. It adds the sum of square of weights for these layers to the loss function. This makes sure that the value of weights remains relatively small during training so that the overall complexity of model does not increase leading to over-fitting.

Class weights: The dataset classes were not equally balanced. This can add a significant bias to our model. In order to prevent it we used class weights to penalise the model more for less represented class. Let X be the vector containing counts of each class X_i where $i \in X$. Then the weights for each class were given as:

$$weight_i = \frac{max(X)}{X_i + max(X)}$$

4 Experimental setup

4.1 Hyper-parameters

We extracted 2048-dimensional image feature vectors from inception which were further reduced to 200 dimensions for fusion step using a dense layer. 100-dimension glove vectors were used for representing meme text. Bi-directional LSTM's of size 200 were used throughout the model. Feature fusion step used a GRU of size 64. A dropout of 0.2 was used on embedding layer. A dropout of 0.4 was used after first LSTM, dropout of 0.1 was used after the fusion step and dropout of 0.2 was used for recurrent connections of GRU and LSTM. L2 regularization of 0.001 is used in the final dense layers. All dense layers used 'relu' activation. The outputs of final layers used a softmax activation.

4.2 Training

We trained all our models to minimize the cross-entropy loss. We used ADAM (Kingma and Ba, 2014) optimiser for backpropagation. Batch size of 200 was used and we trained our models for 120 epochs. Our models were developed on Keras (Chollet and others, 2015) using Tensorflow backend. We trained separate models for each humour class H. The details of branching the model for classification and quantification for each subtask is described as:

Subtask A

The joint feature vector is fed into two different dense layers of three neurons (positive, negative, neutral) and five neurons (very negative, negative, neutral, positive, very positive). For prediction a layer with three neurons was used.

Subtask B and Subtask C

The joint feature vector is fed into two different dense layers of size three and four. Subtask B required us to classify the meme as positive or negative for each class H, where $H \in \{$sarcastic, humorous, offensive, motivational $\}$. Subtask C required us to quantify the humour expressed. The quantification scale was E, where $E \in \{$not, slightly, mildly, very$\}$. For training we used two branches one for quantification meme into scale E. For the other branch we merged the "mild" and "very" semantic into a single class because "very" semantic scale was under-represented for each humour class. We trained separate classifiers for each humour class H. For motivational class quantification scales were not present so we trained it using a single dense layer after joint fusion. For Subtask B prediction, we used the 3 neuron branch and for Subtask C we used the 4 neuron branch.

5 Results and analysis

Our model performed considerably well for Subtask B and Subtask C. Our team rank was 5 and 8. For Subtask A our rank was 23. The final F1 scores for our test set are given in the Table 2.

Subtask	F1(macro)	F1(micro)
A	0.325	0.482
B	0.508	0.612
C	0.313	0.408

Table 2: Results on Test set

We also include our scores on the dev set which was used for submission, it is given in Table 3.

Subtask		F1(macro)	F1(micro)
A		0.758	0.809
B	Humour	0.833	0.885
	Sarcasm	0.832	0.878
	Offense	0.831	0.834
	Motivation	0.829	0.847
C	Humour	0.739	0.759
	Sarcasm	0.733	0.775
	Offense	0.690	0.766
	Motivation	0.829	0.847

Table 3: Results on Dev set

Our model performed considerably well for Subtask B and Subtask C. For Subtask A the lower performance can be due to difficulty in identifying the neutral class from the positive class. These two classes may have similar features which make it difficult for the model in identifying the correct classes. The difference in macro and micro F1 scores for Subtask A justifies the above premise. Another important aspect which we did not cover was the interdependence of humour classes H and sentiment L on each other. We trained independent models for each class for classification and quantification using similar architectures. We used transfer learning using only inception and GloVe models. Testing our architecture with separate models may lead to different results. One of the main problems which we came across was identifying sarcasm in images which is a key component of expressing sentiment and humour using memes. A lot of work has been done in identifying sentiment from text. Identifying sentiment in images that represent sarcasm is a tough task. Combining the two modalities improves the understanding of the memes. But better understanding of sentiments from images may further improve the results in future works.

6 Conclusion

This paper describes our deep learning system that competed in all the subtasks of the SemEval 2020 Task 8: Memotion analysis. Transfer learning was used to extract features from image and text separately, employing the state-of-the-art models. We used LSTM with attention to combine the features of our model. A single model was used for classification as well as quantification for each H, where $H \in \{$sarcastic, humorous, offensive, motivational$\}$. We performed nicely in Subtask B and Subtask C. One important aspect which we would like to work on in the future is the interdependence of humour classes H on each other so that we can create a single model which utilises the interdependence of humour classes and perform classification as well as quantification for each H using a single combined model.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate.

Christos Baziotis, Nikos Pelekis, and Christos Doulkeridis. 2017. DataStories at SemEval-2017 task 4: Deep LSTM with attention for message-level and topic-based sentiment analysis. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 747–754, Vancouver, Canada, August. Association for Computational Linguistics.

Yoshua Bengio, Patrice Y. Simard, and Paolo Frasconi. 1994. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5 2:157–66.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation.

François Chollet et al. 2015. Keras. `https://keras.io`.

Chi Thang Duong, Remi Lebret, and Karl Aberer. 2017. Multimodal classification for analysing social media.

J. H. French. 2017. Image-based memes as sentiment predictors. In *2017 International Conference on Information Society (i-Society)*, pages 80–85.

Laura Graesser, Abhinav Gupta, Lakshay Sharma, and Evelina Bakhturina. 2017. Sentiment classification using images and label embeddings.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November.

Anthony Hu and Seth Flaxman. 2018. Multimodal sentiment analysis to explore the structure of emotions. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '18, page 350–358, New York, NY, USA. Association for Computing Machinery.

Vijayasaradhi Indurthi, Bakhtiyar Syed, Manish Shrivastava, Manish Gupta, and Vasudeva Varma. 2019. Fermi at SemEval-2019 task 6: Identifying and categorizing offensive language in social media using sentence embeddings. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 611–616, Minneapolis, Minnesota, USA, June. Association for Computational Linguistics.

Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization.

Navonil Majumder, Soujanya Poria, Haiyun Peng, Niyati Chhaya, Erik Cambria, and Alexander Gelbukh. 2019. Sentiment and sarcasm classification with multitask learning.

A L V Peirson and E Meltem Tolunay. 2018. Dank learning: Generating memes using deep neural networks.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, October. Association for Computational Linguistics.

Sara Rosenthal, Noura Farra, and Preslav Nakov. 2017. SemEval-2017 task 4: Sentiment analysis in twitter. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 502–518, Vancouver, Canada, August. Association for Computational Linguistics.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252.

Chhavi Sharma, Deepesh Bhageria, William Paka, Scott, Srinivas P Y K L, Amitava Das, Tanmoy Chakraborty, Viswanath Pulabaigari, and Björn Gambäck. 2020. SemEval-2020 Task 8: Memotion Analysis-The Visuo-Lingual Metaphor! In *Proceedings of the 14th International Workshop on Semantic Evaluation (SemEval-2020)*, Barcelona, Spain, Sep. Association for Computational Linguistics.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. 2015. Rethinking the inception architecture for computer vision.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2014. Show and tell: A neural image caption generator.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. SemEval-2019 task 6: Identifying and categorizing offensive language in social media (OffensEval). In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 75–86, Minneapolis, Minnesota, USA, June. Association for Computational Linguistics.