Predicting Sentiments in Image Advertisements using Semantic Relations among Sentiment Labels

Stephen Pilli, Manasi Patwardhan, Niranjan Pedanekar, Shirish Karande
TCS Research, India
(stephen.pilli, manasi.patwardhan, n.pedanekar, shirish.karande)@tcs.com

Abstract

Understanding the sentiments evoked by advertisements is crucial in serving them appropriately to consumers. Advertisements often use images to evoke sentiments. An image can convey multiple sentiments of different nature. Automatically predicting these multiple sentiments can help serve better advertisements to consumers, especially in an online scenario at scale. In this paper, we present a neural network model based on graph convolution to predict such sentiments, which exploits the semantic relationship among the sentiment labels. We use it to predict multiple sentiment labels using an annotated dataset of $30,340$ *image-based advertisements. We also find a distance metric that best represents the distribution of sentiments in the dataset and utilizes it in a loss function that separates applicable sentiments from the non-applicable ones. We report an improvement in mean average precision and overall F1 score over a multi-modal multi-task state-of-the-art model.*

Figure 1: An example of an advertisement conveying 'shocked' as well as 'calm' sentiments at the same time.

1. Introduction

As the adage goes, a picture is worth a thousand words. Numerous companies worldwide profit from selling such thousand words in form of an image advertisement. It is reported that 70.9% of Google's revenue is generated from selling advertisements [2]. Over $1,700$ banner advertisements are served to an average consumer per month, but not more than 50% of them are shown interest by the beholder [13]. This huge gap between the amount of advertisement content available and the efficiency of it being served might be due to lack of a way to automatically understand the advertisement's visual rhetoric [5] at this scale. The persuasiveness of an advertisement can be made efficient by understanding the topic of the ad, but may not be effective unless the emotion conveyed by the advertisement is also understood. Sometimes, not understanding the emotion might end up distancing the consumer. An example would be an advertisement that evokes joy but the context in which the advertisement is served is that of a funeral. Such a mismatch also has the potential to damage a brand's image in the market.

Inferring emotion from an image is not a trivial task. There have been efforts to understand the emotion from the image, one notable attempt being the DeepSentiBank [1]. It is a model trained on the images collected from web that are tagged with multiple sentiments. The reported detectors performance on natural images is notable. Hussain et al. [5] extended DeepSentiBank to advertisement images and found that DeepSentiBank performed poorly in terms of accuracy. They observed that a detector trained on natural images could not be generalized to advertisement images. Besides DeepsentiBank, Vedula et al. [14] developed an advertisement recommendation system using sentiments in multimedia content. Madhok et al. [9] extended the work to advertisement topics, and constructed a unified framework to understand advertisements jointly with sentiments. Zhang et al. [17] proposed a novel deep multi-modal multi-task framework to integrate multiple modalities to achieve effective topic and sentiment prediction simultaneously for advertisement understanding. In this paper, we compare and use the results by Zhang et al. [17] (referred to as the M&M model) as benchmark for sentiment prediction.

In this paper, we try to address certain aspects missing from the previously proposed sentiment prediction frameworks. As observed by Peng et al. [12], an image can invoke emotional ambivalence in a viewer. For instance, in Figure 1, the ad conveys both 'shocked' as well as 'calm' emotion at the same time. We observed this phenomena in the advertisement dataset used by Hussain et al. [5], where one image was tagged by 3.77 sentiments on an average. In a general setting to solve the advertisement sentiment understanding problem, pair-wise semantic relation between the sentiment labels (for example, the semantic relation between 'shocked' and 'calm' in context of the advertisement image in Figure 1) has been typically ignored. In this paper, we use graph convolution over convolutional neural networks to capture such pair-wise semantics. We also try to learn pair-wise relations in the context of visual information extracted from the ad. Such a method was tried by Li et al. [7] in a different setting such as multi-label classification with semantically different labels. But to the best of our knowledge, this is the first time where the semantic relation between sentiments is being exploited in the context of an advertisement. Zhang et al. [17] considered graph convolution neural networks, but have not learnt the semantic relation in context of the advertisements.

Our contributions in this paper are: (1) We analyze the data in an unsupervised manner to find out the distance metric that best fits the sentiment semantics in advertisements, (2) We discuss the loss function based on the proposed metric to exploit semantic relation between sentiment labels. (3) We discuss an architecture based on graph convolution to exploit semantic relation between labels and perform a joint semantic learning within the advertisement context. In addition, we conduct experiments on three datasets derived from the dataset used by Zhang et al. [17] and report a performance improvement in prediction of sentiments. We also compare the loss function based on standard distance metric with the proposed metric and show a performance improvement. Finally, we present the results of an ablation study on the architecture and report their individual contributions.

2. Dataset

The dataset given by Hussain et al. [5] contains $64,140$ advertisement images annotated with topics, sentiments, and action reason pairs. Out of which only $30,340$ are annotated with sentiments. Some examples of advertisements are shown in Figure 2 where each advertisement is annotated with at least one representative sentiment. Besides, the dataset also groups together semantically close sentiments to a representative sentiment as shown in Figure 2 for ease of annotation. For example, the sentiment 'calm' also stands for 'soothed', 'peaceful', 'comforted', 'satisfied' and 'cozy'. Each advertisement in the dataset is annotated by at least 3 raters and at most 5 raters [5]. We present the count of advertisements against each representative sentiment label or the sentiment label distribution in Figure 3.

	Representative	Semantically Related
	calm, fashionable	trendy, elegant, beautiful, attractive, sexy, soothed, peaceful, comforted, satisfied, cozy
	creative, inspired, manly	inventive, productive, motivated, ambitious, empowered, hopeful, determined

Figure 2: Advertisement examples from the dataset annotated with representative sentiments and the corresponding semantically related sentiments as labels.

2.1. Label distribution analysis

The noticeable fact from the Figure 3 is, sentiment label distribution is heavily skewed towards certain labels (in particular, 'creative'), which would cause a sentiment label bias during training of the learning model we propose to employ. This is probably also due to the fact that some annotators interpreted the 'creative' label as the advertisement characteristic rather than the sentiment conveyed by it. Therefore, we proposed an undersampling strategy to solve this problem without reducing the samples in the dataset. The details are described later in the Experiments section.

2.2. Label space analysis

Advertisements in the dataset given by Hussain et al. [5] have been annotated with 115 sentiments. Out of these, 30 sentiments are representative and the rest are semantically related to the representative sentiments. Our proposed idea is to analyse the sentiment label space of the dataset to find the best distance metric that can represent all 115 sentiment labels such that they are semantically close to the representative sentiment label distribution shown in Figure 4 (a). The standard distance metric used in hinge rank loss is $l2-norm$ [3]. To formulate the loss function to best fit our data, we used unsupervised techniques to analyse the sentiment label space. Firstly, we employed agglomerative clustering [11] with number of clusters be-

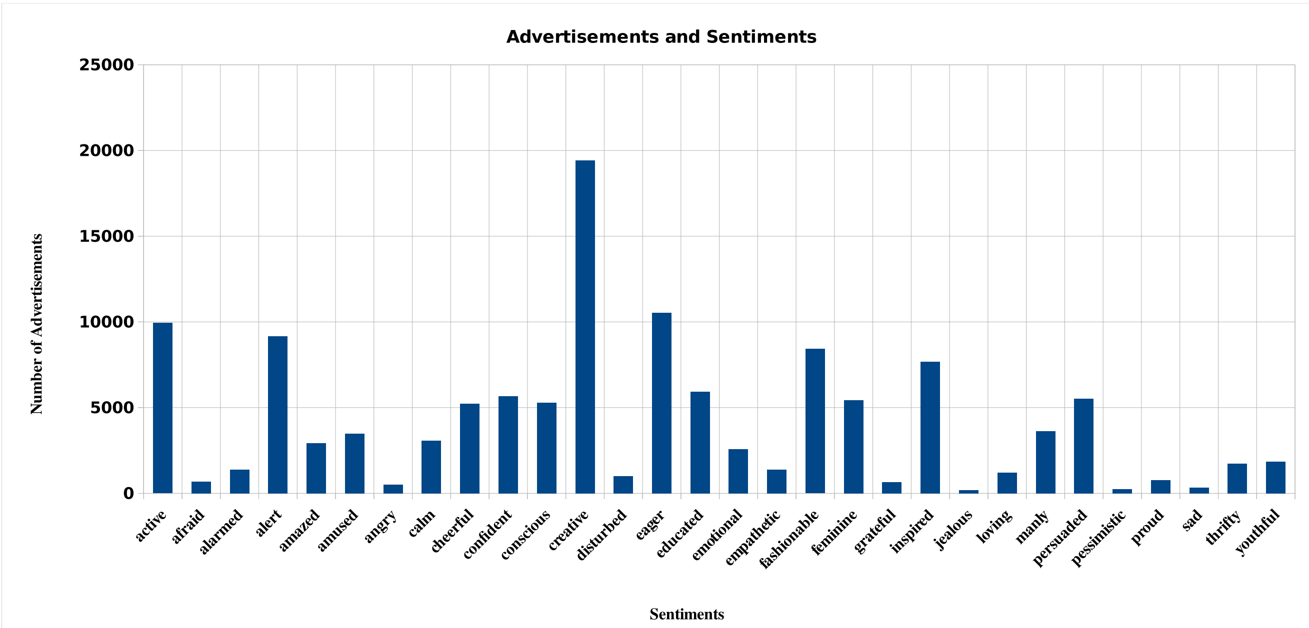
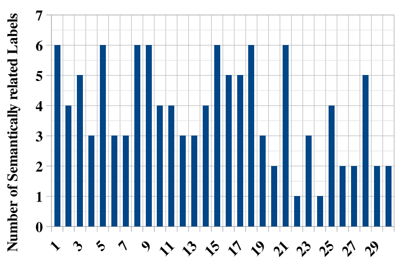
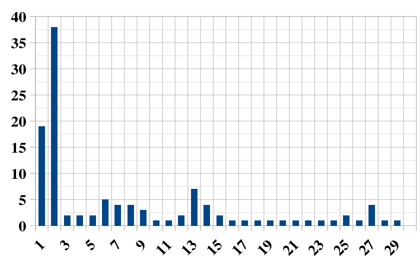


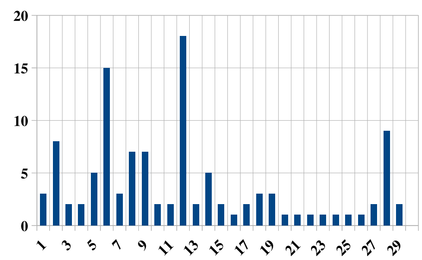
Figure 3: Sentiment label distribution over the entire advertisement dataset.



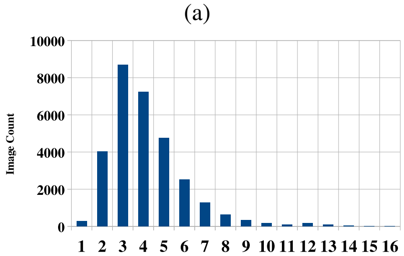
(a)



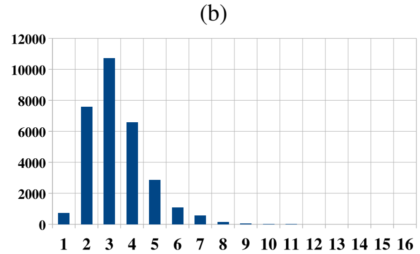
(b)



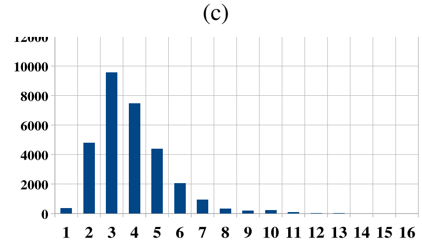
(c)



(d)



(e)



(f)

Figure 4: The x-axis for the top three plots is representative sentiments, and x-axis for the bottom three plots is the number of clusters. Plots (a), (b) and (c) show the ground truth, $l2 - norm$ and $max - norm$ sentiment distributions in the dataset respectively. Similarly, plots (d), (e) and (f) show the data perplexity for ground truth, $l2 - norm$ and $max - norm$ respectively.

ing 30 and linkage set to average. Using this clustering, we clustered the word embedding for the 115 sentiment labels with 15 widely used distance metrics. We observed that $max - norm$ distance metric best fitted this purpose based on two parameters, namely, *sentiment label distribution* and *data perplexity*. The ideal case for first parameter is the sentiment label distribution shown in Figure 4 (a), where the sentiment label clusters are semantically perfect, as they are manually clustered by humans.

In the top plots of Figure 4, we can observe the difference between original sentiment label distribution (a) from the distribution achieved using $l2-norm$ (b), and the distribution clustered with $max-norm$ (c). Though $max-norm$ clusters were semantically weaker compared to original distribution, we considered them to be good as the cluster formation is much closer to the original.

The second parameter, data perplexity, is the image count annotated with more than one sentiment which are semantically different. The ideal scenario for the second parameter is that the number of images per cluster must be close to the dataset's image count per cluster shown in Figure 4 (d). We observed that $max-norm's$ first, second, third, and fourth peaks are closer to original peaks than that of $l2-norm's$. From this, we chose $max-norm$ to formulate the loss function in our model proposed in the following section.

2.2.1 Word embeddings

We used word embedding vectors for each sentiment for clustering them. Word embedding vectors are word representations in continuous vector space, in which semantically similar words fall near each other and dissimilar words fall apart. The skip-gram model proposed in [10] is a widely used and efficient semantic vector representation. In our implementation, we used word2vec with skip-gram model of 300 dimensional word embeddings. Word embeddings for all 115 labels were derived from the Google news corpus.

3. Model architecture

In this section, we provide motivation to use Graph Convolution Network (GCN) along with the formulation of a loss function. We also provide the details of training and inference with the proposed architecture and loss function. The proposed model archictecture is depicted in Figure 5.

3.1. Graph convolution network (GCN)

We employed Graph Convolution Networks [15] to exploit the semantic relations between the sentiments of the advertisements. We also used the model to learn these semantic relations in the context of visual advertisements. A graph convolution neural network is a kind of neural network which operates on graphs. Given a graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$ where, $\mathbf{V}$ represents the vertices and $\mathbf{E}$ represents the edges, $\mathbf{A} \in \mathbb{R}^{n \times n}$ is an adjacency matrix where, $\mathbf{a_{ij}}$ denotes the edge weight between nodes. For every node in the graph, their exists a corresponding $\mathbf{d}$-dimensional feature vector $\mathbf{x}_i \in \mathbb{R}^d$, and the inputs to the nodes $\mathbf{V}$ can be represented as matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$ where, $\mathbf{n}$ is number of nodes and $\mathbf{d}$ is the feature dimension of a each node. Graph convolution for one layer can be represented as

$$\mathbf{G} = \mathbf{A} \cdot \mathbf{X} \cdot \mathbf{W} \tag{1}$$

where $\mathbf{W} \in \mathbb{R}^{d \times m}$ is the weight matrix of this specific layer and $\mathbf{m}$ can be configured based on our usage.

3.2. Visual context in label space semantics

As shown in the architecture in Figure 5, we first learnt to project the d-dimensional feature vector extracted from the convolutional neural network [4] into the sentiment label word embeddings to contextualize each sentiment with the advertisement. The pair-wise similarity between the contextualized sentiments was computed to learn the semantic relation between each sentiment in context of the advertisement. Given an advertisement image I, a feature vector $\mathbf{f}$ was extracted from convolution neural networks where, $\mathbf{f} \in \mathbb{R}^d$. Extracted feature vector $\mathbf{f}$ was learnt to optimally project into embedding matrix $\mathbf{E} \in \mathbb{R}^{n \times k}$ where n is the number of sentiment labels and k is the word embedding dimension to generate a contextualized input to the graph. Equation 2 represents our interpretation mathematically as

$$\mathbf{X} = \sum_{\mathbf{i}} \mathbf{E}_{:,\mathbf{i}} + (\mathbf{T_i} \cdot \mathbf{f}) \tag{2}$$

where, $\mathbf{T} \in \mathbb{R}^{k \times d}$ is the projection matrix learnt during backpropagation and $\mathbf{X} \in \mathbb{R}^{n \times k}$ matrix representation of nodes in the graph. Besides the input representation $\mathbf{X}$, adjacency matrix $\mathbf{A}$ was configured to learn the edge relation of the graph $\mathbf{G}$. We opted to capture pairwise similarity between the nodes to learn the semantics of the sentiments in the context of an advertisement. Equation 3 mathematically represents the computation of adjacency matrix.

$$\mathbf{A} = \mathbf{X^T} \cdot \mathbf{U} \cdot \mathbf{V^T} \cdot \mathbf{X} \tag{3}$$

where, $\mathbf{U} \in \mathbb{R}^{k \times k}$ and $\mathbf{V} \in \mathbb{R}^{k \times k}$ are weights matrices learnt during backpropagation. The edges of the graph $\mathbf{G}$ were normalized using softmax over each row of the adjacency matrix $\mathbf{A}$ before computing graph convolution using Equation 1. The resultant matrix $\mathbf{G} \in \mathbb{R}^{n \times m}$ was used in the loss function that we formulated as described in the next section.

3.3. Loss function

After the graph convolution computation, the resultant graph $\mathbf{G}$ was utilized to compute the loss of the overall system. Based on the conclusions on distance metric detailed in Section 2.2, we employed hinge rank [11] loss with minor modifications in terms of $norm$ and margin m, as shown in Equation 4.

$$\mathbf{L} = \sum_{\mathbf{j}} \mathbf{max}(\mathbf{0}, \frac{\mathbf{1}}{\mid p_i \mid} \sum_{\mathbf{i}} \mid\mid \mathbf{G} \cdot p_i \mid\mid_\infty - \mid\mid \mathbf{G} \cdot n_j \mid\mid_\infty) \tag{4}$$

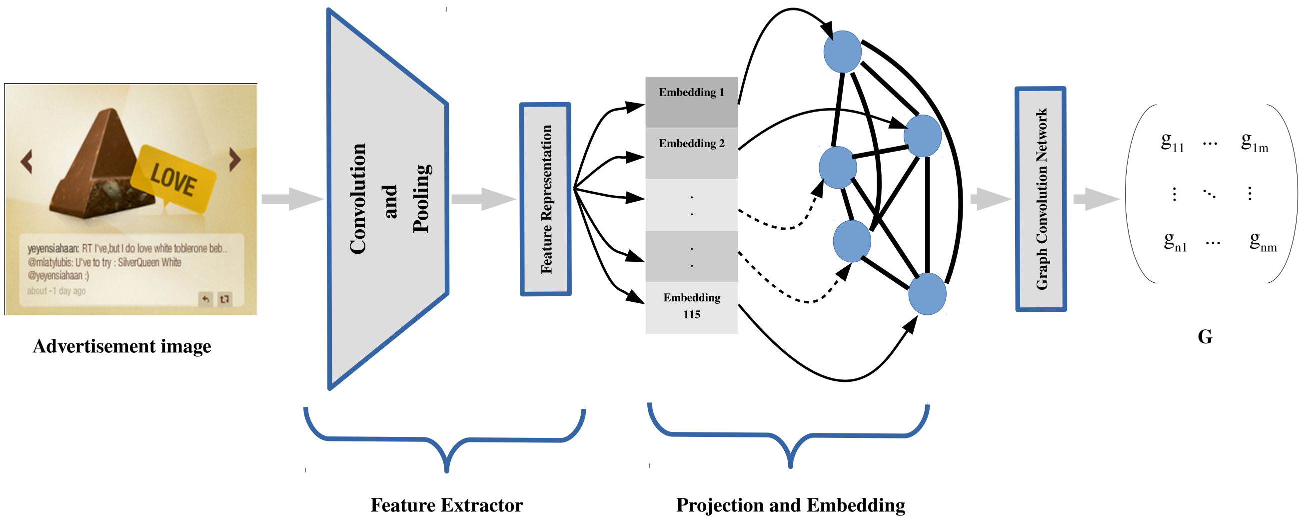


Figure 5: Architecture of the proposed model.

The loss function can be interpreted as the average norm of the resultant graph multiplied with all positive labels that act as a threshold to penalize the violation caused by negative labels. To explain further, if the $max - norm \ || \cdot ||_\infty$ of the resultant graph $\mathbf{G}$ and specific negative sentiment label from the set was less than average $max - norm$ of the resultant graph $\mathbf{G}$ and positive sentiment labels, the system was penalized by the amount of violation. The total violation for the negative labels for an advertisement was summed up and backpropagated. Note that the mechanics of the loss function is exactly same as proposed in [8], but it has been adapted to this specific sentiment label space detailed earlier in Section 2.2.

3.4. Training and inference

We used annotations of each advertisement in the dataset as positive sentiment labels and regarded the rest of the sentiments in the sentiment label space as negative sentiment labels. The matrix $\mathbf{p}$ in Equation 4 contains the word embeddings for positive sentiment labels. Similarly, the matrix $\mathbf{n}$ contains word embeddings for negative sentiment labels.

During training, we provided an extracted feature vector from $resnet - 152$ for each advertisement image as an input to the model giving a matrix $\mathbf{G}$ in Equation 1 at the output. We used $\mathbf{G}$ in the loss function (Equation 4) to train the model end-to-end. We also employed $\mathbf{G}$ to rank the sentiment label set during inference. We initialized the embedding matrix $\mathbf{E}$ in the model with word embeddings of sentiment labels for the training and inference processes. In addition, we also investigated the effect of randomly initializing the embedding matrix versus initializing it with word embeddings in an ablation study discussed later in the paper. We trained the model for 500 epochs with a batch size of 64 and experimented with multiple hyper-parameters to get the best results from the model.

During inference, we matrix-multiplied the model's output for each advertisement image with the sentiment label set $\mathbf{S}$. $\mathbf{S}$ contains column matrices of word embeddings for all 115 sentiments labels. We calculated the distance from the origin for every sentiment label. By sorting the resultant set of distances in an ascending order, we obtained labels which were closest to the origin. The top-k of these labels were considered as predictions for an advertisement image, k being the number of sentiment labels of an advertisement. We used 70% data for training, 20% for validation and 10% for model's testing.

4. Experiments

In this section, we present various experiments conducted with the loss functions formulated using different distance metrics and compare the overall sentiment prediction results with the existing benchmark [17]. We mentioned earlier in Section 2.1 that the skewed distribution of annotated sentiments in favour of a single label ('creative') can influence the prediction results adversely. In this section, we propose four experiments to counter this and present the effect of different distributions of the 'creative' label on overall model performance. Furthermore, we evaluate individual modules in the model along with their effectiveness in contributing to the overall model performance.

4.1. Evaluation protocol

We employed two metrics to evaluate and compare the performance of our model with the model proposed by Zhang et al. [17]. During inference, we compared the top-k predictions against ground truth sentiment set S_i of an advertisement image in the dataset. Each sentiment was weighted, and averaged ratio between the correct predictions and cardinality of the ordered prediction set $\hat{S}_i$ across all images was calculated to measure $mean\ average\ precision$ (mAP). Similarly, recall was calculated as the ratio of correct predictions and $\mid S_i \mid$. To compute weighted average of $precision$ and $recall$, we employed $F1\ score$, a harmonic mean of $precision$ and $recall$. We extensively used mAP and $overall\ F1\ score$ while evaluating the results of our experiments.

4.2. Implementation details

We center-cropped the advertisements in the dataset to $1:1$ ratio and resized them to input dimension of the feature extractor as proposed by He et al. [4]. We applied random horizontal flip transformation and individual channel normalization on all the advertisements in the dataset. We used the neurons in the last fully connected layer of $resnet-152$ [4] as extracted feature representations of advertisement images. We used Google news pre-trained word embeddings to initialize the embedding matrix in the model. We used Adaptive Momentum Estimator (ADAM) optimizer [6] to train our network. The model implementation was done using PyTorch.

As explained in to Section 2.1, the distribution of the 'creative' sentiment label was highly skewed. To examine the effect of an undersampling strategy to counter this, we designed the following two experiments:

- **Experiment 1:** We used the whole dataset with no undersampling strategy.
- **Experiment 2:** Firstly, we found the mean of the distribution and undersampled all the labels randomly which are above two times the mean, which gave us a more uniform distribution. By doing this, the frequency of the most common sentiment label, i.e. 'creative', was reduced to 40% of its original frequency in the dataset. We repeated the experiment 5 times since we had randomly undersampled the 'creative' labels, and reported the average prediction performance in the results.

To understand the advantage of the embedded semantics space (embedding matrix) in our model for predicting unseen labels, we performed the following two experiments:

- **Experiment 3:** We removed the 'creative' label completely from the training and validation datasets. We also removed the three corresponding sentiment labels ('creative', 'inventive', 'productive') from the embedding matrix. We trained the model on this dataset and tested the same on the original test dataset containing the 'creative' label.
- **Experiment 4:** This was similar to experiment 3. The only difference was that we kept the embeddings of the three corresponding sentiment labels ('creative', 'inventive', 'productive') in the embedding matrix.

The test dataset for all the above experiments was kept the same.

Method	mAP	$Overall\ F1$
GCN [17]	0.223	0.339
M&M [17]	0.292	0.453
Our experiment 1	0.327	0.466
Our experiment 2	**0.332**	**0.470**
Our experiment 3	0.325	0.421
Our experiment 4	0.310	0.446
Improvement over GCN	49%	39%
Improvement over M&M	**14%**	**4%**

Table 1: Comparison of sentiment prediction by our model to the state-of-the-art [17]

5. Results and discussion

The evaluation of our model using $max-norm$ as a distance metric in the loss function for the aforementioned four experiments is summarized in Table 1. Our best results are obtained with undersampling the 'creative' label, i.e. with experiment 2, which is an improvement of 14% in mAP and of 4% in $overall\ F1\ score$ over the benchmark [17].

5.1. Effect of undersampling of noisy labels

As seen in the Table 1, the mAP and $F1score$ for experiment 2 improved over experiment 1. As discussed earlier, this improvement was after removing 60% of the 'creative' labels. On examining the $F1\ score$ for prediction of the 'creative' label for experiment 1 (0.785) and 2 (0.774), we did not observe much drop in the same. This shows that removing creative labels did not affect the score of the same. However, the improvement in the $overall\ F1\ score$ from experiment 1 to 2 indicates that there was an improvement in the $F1\ score$ of other labels. A possible explanation is that the 'creative' and its semantically related labels served as noise for some of the other labels, and thus the removal of the noisy label helped in improving the scores of some of the other labels. To understand what type of labels were getting affected by the creative label, we plotted a graph of

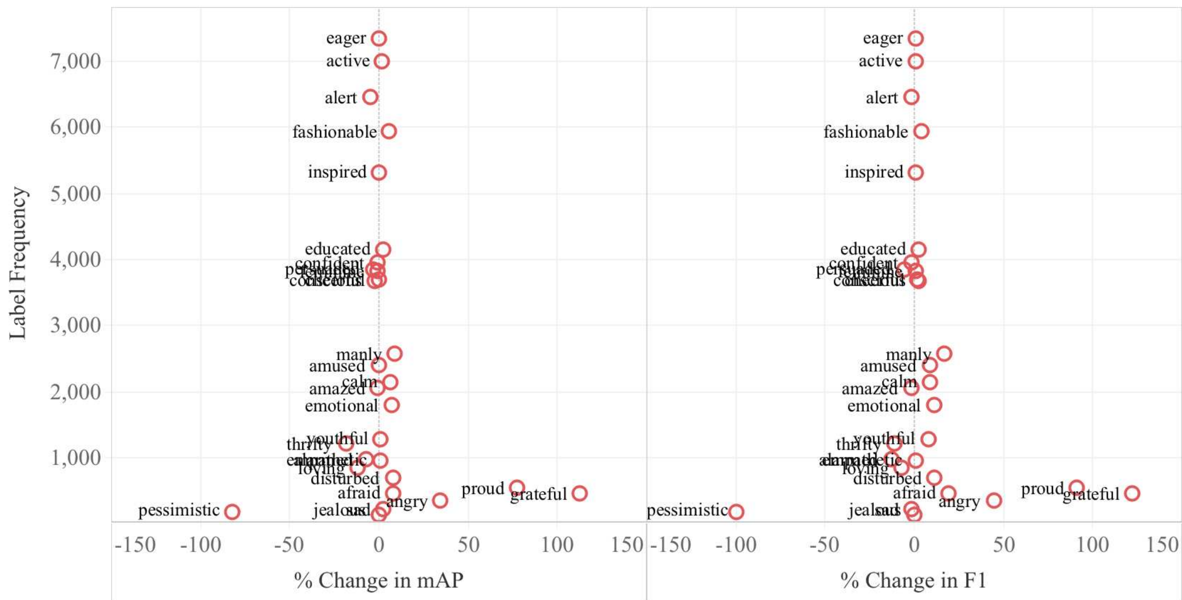


Figure 6: Effect of undersampling of 'creative' label on prediction of labels.

label frequency (in training data) against the % improvement in $F1$ $score$ of the respective label from experiment 1 to 2 as seen in Figure 6. More importantly, we observed that less frequent labels, such as 'proud', 'grateful' and 'pessimistic', were getting affected by the 'creative' label more than the high frequency labels, such as 'active', 'eager' and 'alert'. Some labels such as 'proud' and 'grateful' had a positive effect, whereas, labels such as 'pessimistic' had a negative effect. However, labels having positive effect were more in number and intensity than labels having negative effect, leading to an overall improvement in the $F1$ $score$. Similar observations were made for mAP. We also observed that the effect of the 'creative' label in terms of % change in $F1$ $score$ and mAP for a given label was not in correlation with the semantic similarity of that label to 'creative'.

5.2. Effect of using semantic embedding space

We observed that the $F1$ $score$ for the prediction of the 'creative' label reduced drastically from experiment 1 to 3 (from 0.785 to 0.483), whereas for experiment 4 the drop was comparatively very less (from 0.785 to 0.705). The $overall$ $F1$ $score$ dropped from 0.466 of experiment 1 (where all creative labels were retained) to 0.421 for experiment 3, whereas to 0.446 for experiment 4. This indicates that, for labels which are completely unseen in the training data, but embedded in the semantic space, the performance is better than when the labels are not embedded in the semantic space. Thus, the semantic embedding space is facilitating the learning of unseen labels.

In case of the 'creative' label, when removed from the training data, but included in the semantic space, shows some drop in the results of the label. In such a scenario, the 'creative' label learnt by its semantic embedding (as a part of the semantic space) does not incorporate any noise in the label annotation. As indicated in the earlier sections, the noise could be due to the fact that some annotators interpreted the 'creative' label as the advertisement characteristic rather than the sentiment conveyed by it. Thus, the drop in the performance can be because of the noisy test labels, which have not been predicted as 'creative' by the model.

5.3. The effect of distance metric in loss function

We used the standard $l2-norm$ based loss function and recorded mAP and $overall$ $F1$ $score$ over 500 epochs. Later, we extended the same to the proposed loss function using $max-norm$. We plotted the mAP and $F1$ $score$ at each epoch as shown in Figure 7. From these, it is empirically evident that the distance metric proposed for loss function in Section 2.2 contributes significantly to the overall model performance, as it consistently outperformed the one based on the standard $l2-norm$. Figure 8 reports the percentage performance improvement in mAP and $F1$ $score$ while using $max-norm$ over $l2-norm$. The error is an important parameter in terms of model's convergence rate. Figure 9 shows the error in the model for $l2-norm$ and $max-norm$. We observed that $max-norm$ outperformed $l2-norm$ in all the measurements. From these observations, we concluded that the loss function based on

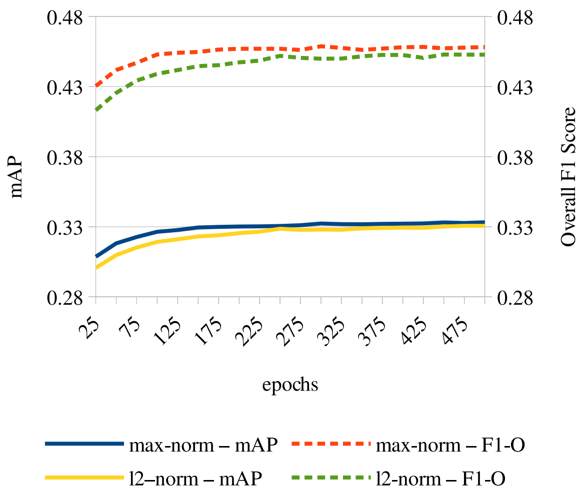


Figure 7: Comparison between the performance of $max-norm$ and $l2-norm$ in terms of mAP and $overall\ F1\ score$.

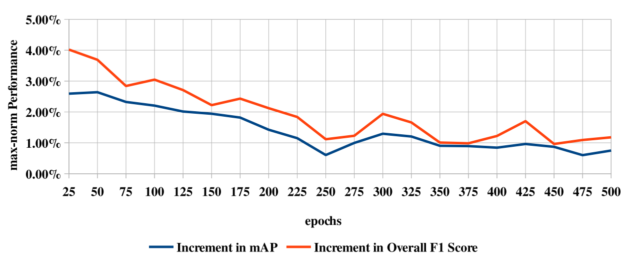


Figure 8: Overall improvement in $max-norm's\ mAP$ and $F1\ score$ over $l2-norm$.

$max-norm$ generalized better than that formulated with $l2-norm$.

5.4. Ablation study

We attempted to understand the effect of adjacency matrix on the overall model performance. We unplugged the adjacency matrix $\mathbf{A}$ from the discussed architecture, and recorded the mAP and $overall\ F1\ score$ for the first 5 epochs. Though the difference is not noticeable initially, as the training progressed, we found the model to perform better with adjacency matrix at higher epochs. Secondly, we attempted to find the effect of random initialization of embedding matrix $\mathbf{E}$ as proposed by Li et al. [7] vis-a-vis initializing $\mathbf{E}$ with word embeddings. We found that initializing $\mathbf{E}$ with the embedding matrix showed 3.64% improvement in mAP and 7% improvement in $F1\ score$ initially. This noticeable improvement gradually decreased with epochs though never converged to zero. So, based on all these observations, we concluded that it is better to initialize $\mathbf{E}$ with word embeddings.

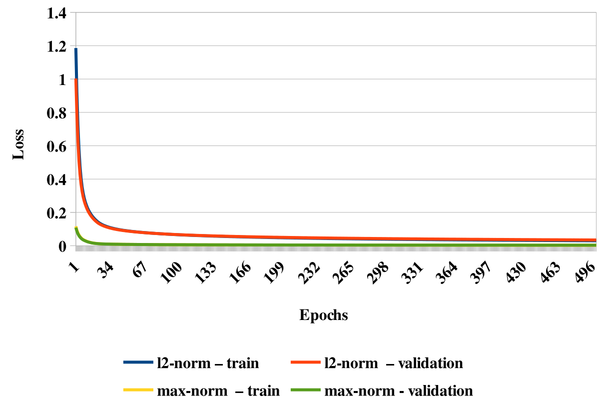


Figure 9: Training and validation error between standard metric and $max-norm$.

6. Conclusion and future work

In this paper, we analysed sentiment label data for advertisements in an unsupervised manner, and proposed a distance metric for the loss function based on the analysis. We reported an improvement over the state-of-the-art results reported by Zhang et al. [17] in predicting sentiments owing to the use of a graph convolutional model along with word embeddings. We discussed the effectiveness of graph convolution in exploiting pair-wise semantic relation in the sentiment labels. We also conducted experiments on datasets derived based on the distribution of the 'creative' sentiment, and reported the contribution of the same to the overall prediction performance. We believe that extending the sentiment labels to a larger space with semantic relationships using word embeddings may find utility in annotating advertisement with correct multiple emotions. We also showed, with limited experimentation, that semantic embedding of unseen labels in our model even in the absence of training data, can help in prediction of these labels.

As future work, we plan to apply our work on unseen labels to practical use-cases using zero-shot learning as prescribed by Xian et al. [16]. We also would like to examine the validity of 'creative' sentiment annotation, and come up with an improved dataset derived from the one used in this paper.

References

[1] Tao Chen, Damian Borth, Trevor Darrell, and Shih-Fu Chang. Deepsentibank: Visual sentiment concept classifica-

tion with deep convolutional neural networks. *arXiv preprint arXiv:1410.8586*, 2014. 1

[2] J. Clement. Google: ad revenue 2001-2018, Feb 2020. 1

[3] Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. In *Advances in neural information processing systems*, pages 2121–2129, 2013. 2

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4, 6

[5] Zaeem Hussain, Mingda Zhang, Xiaozhong Zhang, Keren Ye, Christopher Thomas, Zuha Agha, Nathan Ong, and Adriana Kovashka. Automatic understanding of image and video advertisements. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1705–1715, 2017. 1, 2

[6] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[7] Qiaozhe Li, Xin Zhao, Ran He, and Kaiqi Huang. Visual-semantic graph reasoning for pedestrian attribute recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8634–8641, 2019. 2, 8

[8] Yi-Nan Li and Mei-Chen Yeh. Learning image conditioned label space for multilabel classification. *arXiv preprint arXiv:1802.07460*, 2018. 5

[9] Rishi Madhok, Shashank Mujumdar, Nitin Gupta, and Sameep Mehta. Semantic understanding for contextual in-video advertising. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018. 1

[10] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013. 4

[11] Daniel Müllner. Modern hierarchical, agglomerative clustering algorithms. *arXiv preprint arXiv:1109.2378*, 2011. 2, 4

[12] Kuan-Chuan Peng, Tsuhan Chen, Amir Sadovnik, and Andrew C Gallagher. A mixed bag of emotions: Model, predict, and transfer emotion distributions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 860–868, 2015. 2

[13] Khalid Saleh. Effectiveness of online advertising– statistics and trends, Apr 2018. 1

[14] Nikhita Vedula, Wei Sun, Hyunhwan Lee, Harsh Gupta, Mitsunori Ogihara, Joseph Johnson, Gang Ren, and Srinivasan Parthasarathy. Multimodal content analysis for effective advertisements on youtube. In *2017 IEEE International Conference on Data Mining (ICDM)*, pages 1123–1128. IEEE, 2017. 1

[15] Yuexin Wu, Yichong Xu, Aarti Singh, Yiming Yang, and Artur Dubrawski. Active learning for graph neural networks via node feature propagation. *arXiv preprint arXiv:1910.07567*, 2019. 4

[16] Yongqin Xian, Christoph H Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly. *IEEE transactions on pattern analysis and machine intelligence*, 41(9):2251–2265, 2018. 8

[17] Huaizheng Zhang, Yong Luo, Qiming Ai, Nana Hou, and Yonggang Wen. Look, read and feel: Benchmarking ads understanding with multimodal multitask learning. *arXiv preprint arXiv:1912.10248*, 2019. 1, 2, 5, 6, 8