10-2017

Visual sentiment analysis for review images with item-oriented and user-oriented CNN

Quoc Tuan TRUONG
Singapore Management University, qttruong.2017@phdis.smu.edu.sg

Hady Wirawan LAUW
Singapore Management University, hadywlauw@smu.edu.sg

DOI: https://doi.org/10.1145/3123266.3123374

Citation

TRUONG, Quoc Tuan and LAUW, Hady Wirawan. Visual sentiment analysis for review images with item-oriented and user-oriented CNN. (2017). *MM '17: Proceedings of the ACM Multimedia Conference, Mountain View, CA, October 23-27*. 1274-1282. Research Collection School Of Information Systems.
Available at: https://ink.library.smu.edu.sg/sis_research/3885

Published in MM '17: Proceedings of the ACM Multimedia Conference, Mountain View, CA, October 23-27, Pages 1274-1282
https://doi.org/10.1145/3123266.3123374

Visual Sentiment Analysis for Review Images with Item-Oriented and User-Oriented CNN

Quoc-Tuan Truong
Singapore Management University
80 Stamford Road
Singapore 178902
qttruong.2017@smu.edu.sg

Hady W. Lauw
Singapore Management University
80 Stamford Road
Singapore 178902
hadywlauw@smu.edu.sg

ABSTRACT

Online reviews are prevalent. When recounting their experience with a product, service, or venue, in addition to textual narration, a reviewer frequently includes images as photographic record. While textual sentiment analysis has been widely studied, in this paper we are interested in visual sentiment analysis to infer whether a given image included as part of a review expresses the overall positive or negative sentiment of that review. Visual sentiment analysis can be formulated as image classification using deep learning methods such as Convolutional Neural Networks or CNN. However, we observe that the sentiment captured within an image may be affected by three factors: image factor, user factor, and item factor. Essentially, only the first factor had been taken into account by previous works on visual sentiment analysis. We develop item-oriented and user-oriented CNN that we hypothesize would better capture the interaction of image features with specific expressions of users or items. Experiments on images from restaurant reviews show these to be more effective at classifying the sentiments of review images.

KEYWORDS

visual sentiment analysis; convolutional neural networks; review images

1 INTRODUCTION

Online reviews are fast becoming one of the primary ways to evaluate a multitude of options. For instance, we may look up Amazon reviews when deciding which product would best meet our particular purpose. When on the move, we may check out Yelp reviews while picking a place to have a meal. The usefulness of reviews is derived from their role in capturing the experiences of previous consumers well. In particular, one key piece of information we seek to detect in reviews is the expressed sentiment by the consumer, ultimately whether she had had a positive or negative experience. Inferring sentiments is a critical and fundamental task for review analysis, as sentiments may reveal the preferences of users, as well as the strengths and weaknesses of items. Such information is valuable for recommendation, product design, marketing, etc.

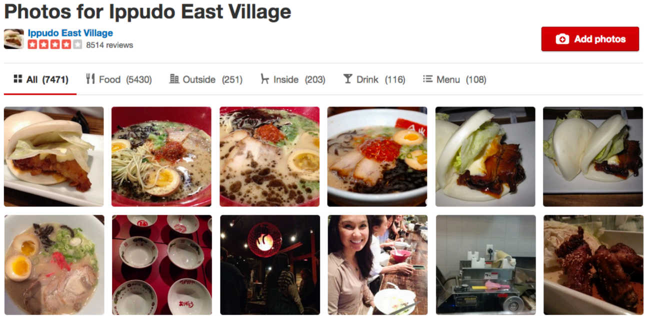

Figure 1: Example Photos for Ippudo East Village at Yelp

The vast majority of past research on sentiment analysis have focused on textual data [22]. In the early days of the Web when online reviews were born, most Web content were textual. However, today's Web is richly multimedia, and text is not the only form of self-expression. Another proliferating form of self-expression is photography [9]. The Web is now awash with visual imagery. The popularity of Instagram is one such manifestation[1]. Another manifestation of this trend is the inclusion of images in online reviews. If the purpose of a review is to capture one's experience as vividly as possible, what better way than to do so photographically.

For instance, let us take a restaurant in New York City (NYC) by the name of *Ippudo East Village*[2]. Figure 1 shows some example photos taken by its reviewers. These photos concern various aspects of the reviewers' experience, including food, outside view, inside ambience, drinks, and menu. As of the time of writing, *Ippudo East Village* has 8514 reviews and 7471 images on Yelp, or approaching an average of one image per review. For another example, another highly-reviewed restaurant in NYC by the name of *Traif*[3] (images not shown due to space limitation), there are even more images (2120) than reviews (1652). Keep in mind that this number includes reviews as far back as 10 years ago with few, if any, images. More recent reviews are expected to include more images.

Problem. Given the escalating use of visual imagery in online reviews, we investigate to what extent we could detect the sentiment expressed by the photos included in a review[4]. We deem the overall rating of a review to be a close proxy to the sentiment expressed by the review. For each review image, we seek to detect the sentiment of the review (positive or negative).

[1] http://blog.instagram.com/post/146255204757/160621-news
[2] https://www.yelp.com/biz/ippudo-east-village-new-york
[3] https://www.yelp.com/biz/traif-brooklyn
[4] To keep the focus solely on images, in this study, we do not make use of the review text. Multi-modal sentiment analysis would be an interesting future work.

Our work investigates visual sentiment analysis for *review images*. Previous work in visual sentiment analysis [3, 24] applied it to Flickr images, with sentiment labels based on tags. [29] applied it to Twitter images, including facial recognition. While the core concern is still about images, we hypothesize there might be subtle differences between review images and other types of social media images. In Flickr, for instance, the origin of an image is not always clear. They may also be drawings, not always photography. Most review images are genuine photography taken by the reviewers themselves. They capture various aspects (e.g., food, cleanliness, value), and not just facial expressions. Moreover, some images in social media, e.g., Twitter, may be part of memes [27], which might have been doctored and designed to evoke sentimental reaction.

Approach. We particularly focus on the observation that the sentiment expressed by a review image is likely influenced by three factors: *image factor* (sentiment encoded in the image itself); *user factor* (sentiment expressed by a reviewer through an image); and *item factor* (sentiment associated with an image due to an item). As we survey in Section 2, previous works have relied primarily on image factor alone, associating the sentiment inherently with the image itself, effectively assuming that an image is either universally positive or negative. We postulate that sentiments in online reviews are by nature relative. A piece of furniture may look retro in one restaurant, but may look run-down in another (item factor). A reviewer may find the ambience of a newly renovated place clean and elegant, while another may find the same sterile and uninspired (user factor). The question of sentiment expressed by an image may be inseparable from the idiosyncratic preferences of the reviewer, as well as the peculiar natures of the item or place being reviewed.

Recent approaches for image classification rely on deep learning, such as Convolutional Neural Networks (CNN). AlexNet [17] is one well-known such model, which inspires our base model for Visual Sentiment Convolutional Neural Networks, which we refer to as VS-CNN. Importantly, this base model may not be equipped to deal with the relative preferences of reviewers (user factor) or the peculiar characteristics of items being reviewed (item factor). To take item factor into account, we go beyond the base model and propose an *item-oriented* model or *i*VS-CNN, which incorporates item-specific parameters, so as to reflect how some image features are interpreted in the context of that item. Correspondingly, to reflect user factors, we build a *user-oriented* model or *u*VS-CNN, which incorporates user-specific parameters, so as to reflect how some image features are interpreted through the lens of that user.

Contributions. This work makes the following contributions. First, to our best awareness, this is the first work to study visual sentiment analysis for better understanding of review images. We review the previous works, and present our contrasts in Section 2.

Second, to deal with potentially relevant item- and user-factors in detecting the sentiments of review images, we develop item-oriented *i*VS-CNN and user-oriented *u*VS-CNN in Section 3. Moreover, as CNNs have various types of internal components: convolutional layers and fully-connected layers, it is not clear beforehand which would be the more appropriate component to associate with the item- or user-orientation. We systematically study both types of orientation. We describe the learning details in Section 3.4.

Third, in Section 4, we conduct a comprehensive set of experiments to evaluate the effectiveness of the base model VS-CNN, the item-oriented model *i*VS-CNN, and the user-oriented model *u*VS-CNN. Experiments on real-life image dataset from Yelp.com covering 7 major US cities show that item- and user-oriented models respectively outperform the base model. Incorporating the orientation at higher levels of abstraction seems particularly promising.

2 RELATED WORK

Visual Sentiment Analysis. Sentiment analysis was pioneered for text [2, 6, 12, 15, 21, 22, 25]. Visual sentiment analysis deals with classifying the polarity of an image. One way is to represent an image in terms of color and SIFT features, and then employing classification algorithms such as SVM or Naive Bayes [24, 29]. Another way is to feed the image into a deep learning framework such as CNN [3, 4, 27]. Our work builds on the framework of CNN, with several key distinctions. First is the difference in the types of images. Previous works [3, 4, 27] train on social media images from Flickr, labeling their polarity based on tags. It also effectively assumes that the sentiment of an image can be captured by the tags alone. In contrast, we focus on review images. Second is the difference in the CNN architecture. Previous works use CNN with globally shared parameters, whereas we investigate item and user factors respectively to see their potential effects on visual sentiment analysis. By focusing on images alone, our work is also different from those that focus on bridging two modalities, such as text (e.g., captions, tags) and images [5, 26, 28]. By focusing on review images, which may be diverse, our work is neither limited to, nor especially geared for recognizing human facial expressions [1].

Visually-Aware Recommender Systems. Recommender system estimates how much a user would like an item. It is commonly formulated as rating prediction using matrix factorization [16]. It has been observed that images have a role in e-commerce [7, 8]. When an item image is available, it could be used as additional feature. A user's preference for an item is "transferred" to other items with "similar" images [10, 11]. Though sentiment analysis is potentially useful for recommendation, it is fundamentally a different problem. Recommendation models the relationship between a user and an item. The key information is derived from which items the user has liked previously, and an item is usually associated with only one representative image [10, 11]. In contrast, sentiment models the polarity of an image itself. In our models, though some parameters may be item- or user-oriented, what is learned is the mapping between image features to the sentiment. Yet another form of visually-aware recommendation [14] is to recommend products similar to a photo, essentially an image retrieval problem.

3 CNN FOR VISUAL SENTIMENT ANALYSIS

CNN has been successfully used in learning tasks such as handwriting recognition [18], document recognition [19], feature learning [20], sentence classification [15], image classification [17].

In essence, visual sentiment analysis is an image classification task. As the utility of CNN for a problem is related to its architecture, we investigate how user and item factors could be incorporated into the architecture of CNN. In the following, we first describe a base CNN architecture for visual sentiment analysis, which we refer to as VS-CNN. Then, we illustrate both the item-oriented *i*VS-CNN and the user-oriented *u*VS-CNN respectively.

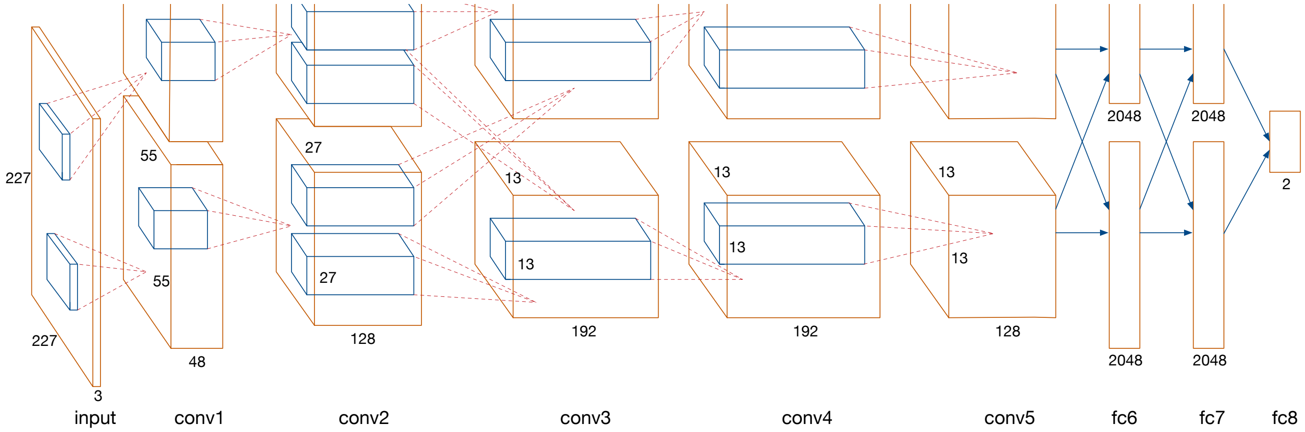

Figure 2: Base CNN Model for Visual Sentiment Analysis: VS-CNN

3.1 Base Model: VS-CNN

CNN involves convolutional (*conv*) and fully-connected (*fc*) layers. Convolutional layers learn features based on spatial correlations in the data. The first convolutional layer learns low-level features from the input images. The next learns higher-level features from the features learned in the previous layer. Eventually, the final convolutional features feed into fully-connected layers that conduct the high-level reasoning of mapping these features into classes.

Our focus is on investigating item and user-orientation. We opt to start from a reasonable base CNN model, beginning with the architecture from AlexNet [17], the winner of ImageNet Large Scale Visual Recognition Challenge 2012 (ILSVRC-2012). The ImageNet task classifies an image into one of 1000 classes. Visual sentiment analysis only considers 2 classes (positive and negative), but as also noted by [27], it may be more challenging as sentiment analysis is probably a higher abstraction than object recognition, as the former needs to be learnt from many images involving similar objects.

Figure 2 shows the architecture of the base model VS-CNN. Input images are preprocessed and cropped to 227 x 227 pixels. Like AlexNet, it has five convolutional layers (*conv1* to *conv5*) and three fully-connected layers (*fc6* to *fc8*). The first, second and fifth convolutional layers are followed by max-pooling layers. There are normalization layers after the first two pooling layers. The two streams of convolutional and fully-connected layers are designed to accommodate learning using 2 GPUs if so desired. Unlike AlexNet, the last fully-connected layer *fc8* now has only 2 neurons for positive and negative sentiment classes, instead of the original 1000. This base model is slightly different to the CNN architecture used in previous works on visual sentiment analysis [4, 27], with variations in the number of layers and neurons, and in its two-stream design.

While the target images from reviews in our visual sentiment analysis dataset are not identical to ImageNet, they are less numerous than the ImageNet collection. Hence, a further advantage to using AlexNet as a base architecture is the pre-trained parameters that AlexNet has extracted from 1 million annotated images from ILSVRC-2012 dataset. Our approach is to initialize our model with the pre-trained referenced model by BLVC, and fine-tune the model parameters inside Caffe [13], a deep learning framework for images.

3.2 Item-oriented Model: *i*VS-CNN

The base model assumes that sentiment is purely a function of the image features. That may hold for images universally considered positive or negative, e.g., an image of a dirty toilet. However, there could be other images that connote positively for some items, but negatively for other items. For instance, an image of people lining up may imply popularity (positive), or slow service (negative).

We hypothesize that there is item-specific factor that would help identify the sentiment of an image. We propose to fine-tune the model, by allowing an item to have its own specific parameters, while sharing most of the parameters with other items. What is not clear is where these item-specific factors are to be incorporated into the CNN. We systematically investigate two logical ways.

3.2.1 Realizing Orientation in Convolutional Layer. One hypothesis is that items extract spatial features differently. To investigate this, we introduce item-orientation into one of the convolutional layers, by dedicating k filters to encode the item orientation. This is illustrated in Figure 3 (best seen in color). A particular convolutional layer has two equi-sized blocks. Each block has n filters, e.g., *conv1* has $n = 48$ filters. Out of the n filters that make up a block, $\frac{k}{2}$ filters (colored red) are made specific to an item, while $(n - \frac{k}{2})$ filters (colored grey) are shared among all items. With the two-stream design, this still results in k item-specific filters in total. The modification ensures that those filters can still be in touch with all features from the previous layer and can be used to learn features of the following layer. This allows an item to pick up some spatial features not necessarily picked up by other items.

There are further questions regarding how many filters are item-oriented, and at which level of abstraction. To investigate the former, we experiment with $k = 8$ and $k = 16$. In general, larger k is appropriate when there is greater differentiation among items. To investigate the latter, we experiment with different convolutional layers, i.e., low (*conv1*), mid (*conv3*), high (*conv5*). When we introduce the specific filters in the first layer (*conv1*), this would capture low-level features. When we introduce them in subsequent layers, they would capture correspondingly higher-level features. Note that the number of filters is still the same as the base model, though now we have item-specific parameters for the k filters.

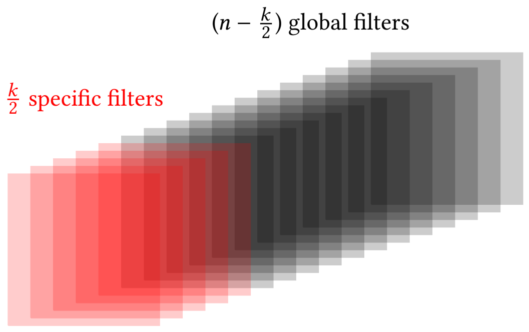
$(n - \frac{k}{2})$ global filters
$\frac{k}{2}$ specific filters

Figure 3: Realizing Orientation in Convolutional Layer

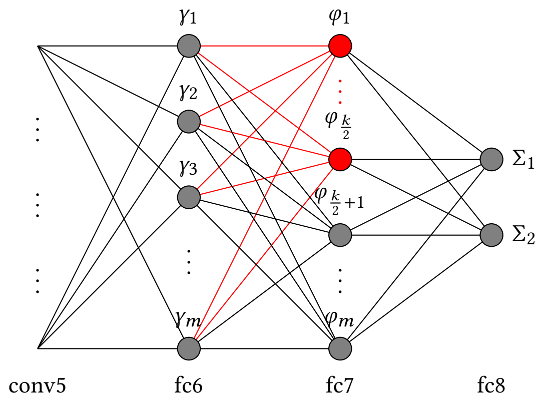

Figure 4: Realizing Orientation in Fully-Connected Layer

3.2.2 Realizing Orientation in Fully-Connected Layer. Another hypothesis is that items extract similar spatial features, i.e., share similar convolutional layers, but they reason with those spatial features differently. To investigate this, we introduce the item-orientation in one of the fully-connected layers. There are two fully-connected layers (*fc6* and *fc7*) just before the final sentiment classification layer (*fc8*). While we could introduce the item-orientation in either layer, to represent the highest level of abstraction, in this work we illustrate the modification of the penultimate layer *fc7*.

This adjusted architecture is now shown in Figure 4. Each stream of the *fc7* layer has $m = 2048$ neurons. Of these, we make $\frac{k}{2}$ neurons: $\varphi_1, ..., \varphi_{k/2}$ (colored red) item-specific, while the other $m - \frac{k}{2}$ neurons (colored grey) remain globally shared. Taking into account both streams, we have k item-specific neurons in total. Similar to modeling orientation in the convolutional layer, we will experiment with $k = 8$ and $k = 16$ in the fully-connected layer.

In summary, the item-oriented model *i*VS-CNN effectively allows different modes of item-orientation, simulating the continuum of increasing level of abstraction along the CNN architecture. The lowest level of abstraction is modeled by orienting the very first convolutional layer. The highest is modeled by orienting the fully-connected layer right before the final classification layer.

3.3 User-oriented Model: *u*VS-CNN

Just as we could model item-orientation into the CNN architecture for visual sentiment analysis, we could also model user-orientation into the CNN architecture in a symmetrical way. To some extent, we seek to capture expressions of sentiments that may be subjective or user-dependent. The user-orientation is also modeled by using either user-specific filters in a convolutional layer, or user-specific neurons in a fully-connected layer.

Though it may be perceived to be the logical next step, we stop short from incorporating both user-orientation and item-orientation simultaneously. That would have assumed that a user-item pair could be associated with images of different sentiments. For online reviews, a user rates an item just once. Hence, doing that potentially models the interaction of users and items directly while bypassing the role of image features (essentially turning it into a recommendation problem). Here, we seek to concentrate on the interaction of image features and either user or item-factors towards visual sentiment, and thus we model user or item respectively.

3.4 Learning Details

For learning the models, we minimize cross-entropy classification loss over softmax output class probabilities by stochastic gradient descent. We begin by discussing the base model VS-CNN. We train it with a batch size of 50 images, momentum of 0.9, and weight decay of 0.0005. Parameters are initialized from the pre-trained model of BVLC inside Caffe [13] framework. We run total 100,000 iterations. The update rule for weight $[w]$ is as follows:

$$v_{i+1} := 0.9.v_i - 0.0005 \cdot \epsilon \cdot w_i - \epsilon \cdot \left\langle \frac{\partial L}{\partial w}\bigg|_{w_i} \right\rangle_{D_i}$$

$$w_{i+1} := w_i + v_{i+1}$$

where i is the iteration index, v is the momentum variable, ϵ is the learning rate, and $\left\langle \frac{\partial L}{\partial w}\big|_{w_i} \right\rangle_{D_i}$ is the average over the ith batch D_i of the derivative of the objective with respect to w, evaluated at w_i.

For both *i*VS-CNN and *u*VS-CNN, there are a couple of challenges in realizing the orientation in convolutional or fully-connected layer. First is the difference in batch size. For *i*VS-CNN, a batch should include only images of the same item as we need to update the item-specific parameters. Naively using a batch size of one makes the learning process unstable. To deal with this, we modify the architecture slightly by not using dropout regularization technique, which helps the models in converging. In addition, we reduce the momentum to 0.5, because the specific neuron/filter parameters are changing to a different set corresponding to a spefic item (or user) in each iteration, and it is less sensitive to previous iterations than the global parameters. The update rule for weight $[w]$ becomes:

$$v_{i+1} := 0.5.v_i - 0.0005 \cdot \epsilon \cdot w_i - \epsilon \cdot \frac{\partial L}{\partial w}\bigg|_{w_i}$$

$$w_{i+1} := w_i + v_{i+1}$$

Second, to learn help the item-specific (or user-specific) parameters, in the first one-fifth of the iterations, we only update those parameters, while keeping all the other parameters stable. This provides a good start for the specific neurons/filters, before further fine-tuning the whole model for the remaining iterations.

4 EXPERIMENTS

The objectives are better understanding of visual sentiment analysis for review images, and investigation of the impact of item-orientation and user-orientation in the CNN architecture. We will first delve into item-orientation, before going into user-orientation.

4.1 Experiments with Item-oriented CNN

Dataset. We use a dataset of review images crawled from *Yelp.com*, covering businesses in 7 different US cities: Boston, Chicago, Houston, Los Angeles, New York, San Franscisco, and Seattle. We derive the sentiment classes from ratings. Each review has a rating from 1 to 5. Ratings 1 and 2 are considered negative, 3 neutral, while 4 and 5 positive. This conversion is similar to previous works [23]. We concentrate on discriminating between positive and negative sentiments only. All images in a review are assigned the same label. We create a balanced dataset where each item (business) has the same number of positive and negative images. As there are more positive than negative images, we retain all of the latter, and sample the same number of the former via stratified random sampling. This dataset has 96,846 images involving 8,318 different businesses and 27,676 users. On average, each item has 11.6 images from 6.7 users. We sample 80% for training and 20% for testing for each item.

Metrics. The task is to classify an image into positive or negative. Each model outputs the probability of positive class for a test image. We employ three classification metrics to evaluate their outputs.

The first metric is **Pointwise Accuracy**. For a test image i, the model outputs its probability $\hat{p}_i \in [0, 1]$ of being in the positive class. The predicted label $\hat{y}_i$ is positive (1) if $\hat{p}_i \geq 0.5$, and negative (0) otherwise. This metric evaluates the number of correct predictions over the total number of testing instances, as defined below.

$$pointwise\_accuracy(\hat{y}, y) = \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}(\hat{y}_i, y_i)$$

where N is number of testing instances, $\hat{y}_i \in \{0, 1\}$ is the predicted label of instance i, $y_i \in \{0, 1\}$ is the corresponding true label, and $\mathbb{1}$ is the indicator function defined as:

$$\mathbb{1}(\hat{y}_i, y_i) = \begin{cases} 1 & \text{if } \hat{y}_i = y_i \\ 0 & \text{otherwise.} \end{cases}$$

The second metric is **Pairwise Accuracy**, which tests the ability of a model to assign a higher probability for a true positive than for a true negative. This is defined as follows.

$$pairwise\_accuracy(\hat{y}, y) = \frac{1}{M} \sum_{k=1}^{M} \delta(\hat{z}_{i,j}, z_{i,j})$$

where $M = N/2$ is the number of pairs of testing instances. Each pair consists of one positive image and one negative image randomly selected from images of the same item. $\hat{z}_{i,j} = (\hat{p}_i, \hat{p}_j)$ is the predicted probabilities for a pair of instances (i, j). $z_{i,j} = (y_i, y_j)$ is the corresponding true pair of labels. δ is a function defined as:

$$\delta(\hat{z}_{i,j}, z_{i,j}) = \begin{cases} 1.0 & \text{if } \hat{p}_i > \hat{p}_j \text{ and } y_i > y_j \text{ (same ranking order)} \\ 1.0 & \text{if } \hat{p}_i < \hat{p}_j \text{ and } y_i < y_j \text{ (same ranking order)} \\ 0.5 & \text{if } \hat{p}_i = \hat{p}_j \text{ (break ties at random)} \\ 0.0 & \text{otherwise.} \end{cases}$$

The third metric is **Mean Absolute Error (MAE)**, which returns the average difference between the actual label value and the predicted probability. The lower the error, the better the model is.

$$MAE(\hat{p}, y) = \frac{1}{N} \sum_{i=1}^{N} |\hat{p}_i - y_i|$$

Quantitative Evaluation. We now look at the sentiment classification results. First, we discuss the results based on pointwise accuracy in Table 1. As this is a balanced dataset, a random classifier is expected to achieve an accuracy of 0.5. As a reference baseline, we include Naive Bayes (NB) classifier trained with features extracted from the penultimate layer of AlexNet. The NB classifier can benefit from good image representation from a state-of-the-art CNN model, achieving pointwise accuracy of 0.54. The base model VS-CNN that learns a global classification function achieves pointwise accuracy of 0.54 which is comparable to NB. That these are higher than random is itself interesting, implying that there are some information signals within review images that provide cues to the overall sentiment of the review. Though we learn a single model, we show detailed results for each city, and the accuracy results are quite consistent across all the cities.

We now investigate whether the item-orientation has an effect. As discussed in Section 3.2, there are several ways of introducing item-orientation to *i*VS-CNN. *LowConv* refers to modeling it in the lowest level of abstraction, by incorporating item-specific filters in the first convolutional layer (*conv1* in Figure 2). Notably, this increases the pointwise accuracy to around 0.56. Moving the item-orientation to the middle *MidConv* (*conv3*), and then to the high abstraction level *HighConv* (*conv5*), results in further increases in accuracy to around 0.61. While there is not much difference between *MidConv* and *HighConv*, modeling item-orientation in the fully-connected layer *FC*, which is the highest abstraction just before sentiment classification, results in the highest accuracy of around 0.62. Best performance in each row is boldened. These results provide supporting evidence to two points. First, there are slight variances across items when modeling visual sentiments, and taking those into account results in higher accuracy. Second, the item-orientation seems to be a high-level concept that is better modeled at higher levels of feature abstraction.

In turn, Table 2 provides the corresponding results in terms of pairwise accuracy. In general, it supports the previous observations. Of additional note is the slightly higher accuracy numbers in general, as compared to Table 1. This implies that even in those cases that the model may not assign appropriate probability values in absolute terms, the relative rankings between positive and negative classes are better preserved. Finally, Table 3 shows the results in MAE. Unlike the previous two tables on accuracies, here lower errors are better. MAE is sensitive to how far away the predicted probabilities are from the correct labels. In this respect, the item-oriented *i*VS-CNN models are also better than the base model and NB as well.

Thus far, the results we have discussed are for $k = 16$. To see if varying k has much effect on the results, we produce a summary of the average results for various metrics for both $k = 8$ and $k = 16$ in Table 4. Evidently, $k = 16$ achieves better results than $k = 8$. Notably, even at $k = 8$, the results for *i*VS-CNN are still better than

Table 1: Item-oriented – Pointwise Accuracy (higher is better)

City	NB	VS-CNN	*i*VS-CNN			
			LowConv	MidConv	HighConv	FC
Boston	0.526	0.542	0.563	0.614	0.615	**0.629**
Chicago	0.554	0.540	0.558	0.631	0.625	**0.633**
Houston	0.549	0.546	0.557	0.612	0.619	**0.620**
Los Angeles	0.537	0.547	0.561	0.601	0.603	**0.615**
New York	0.526	0.541	0.568	0.606	0.609	**0.621**
San Francisco	0.550	0.546	0.567	**0.623**	0.620	0.619
Seattle	0.542	0.542	0.563	0.591	**0.601**	**0.601**
Avg.	0.539	0.544	0.563	0.610	0.612	**0.620**

Table 2: Item-oriented – Pairwise Accuracy (higher is better)

City	NB	VS-CNN	*i*VS-CNN			
			LowConv	MidConv	HighConv	FC
Boston	0.528	0.583	0.593	0.659	0.666	**0.686**
Chicago	0.557	0.588	0.608	0.673	0.688	**0.697**
Houston	0.569	0.592	0.599	0.651	**0.667**	0.662
Los Angeles	0.552	0.561	0.577	0.645	0.647	**0.672**
New York	0.540	0.566	0.593	0.650	0.657	**0.673**
San Francisco	0.563	0.569	0.603	0.666	0.672	**0.688**
Seattle	0.552	0.577	0.594	0.653	0.636	**0.676**
Avg.	0.551	0.572	0.592	0.655	0.660	**0.678**

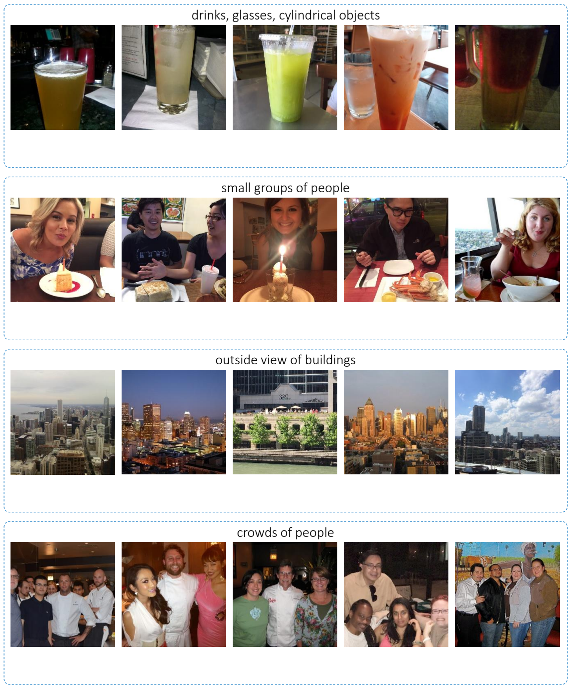
drinks, glasses, cylindrical objects
small groups of people
outside view of buildings
crowds of people

Figure 5: Item-oriented – Most positive images from the base model VS-CNN.

the base model VS-CNN (unaffected by k). Our goal is to investigate the effect of item-orientation, which is shown by both $k = 8$ or $k = 16$; it is not our intention to delve into the best settings of k.

Table 3: Item-oriented – MAE (lower is better)

City	NB	VS-CNN	*i*VS-CNN			
			LowConv	MidConv	HighConv	FC
Boston	0.473	0.487	0.439	0.388	0.385	**0.372**
Chicago	0.446	0.487	0.437	0.373	0.373	**0.368**
Houston	0.452	0.486	0.443	0.387	**0.382**	0.383
Los Angeles	0.464	0.489	0.445	0.399	0.399	**0.387**
New York	0.474	0.493	0.436	0.396	0.394	**0.381**
San Francisco	0.450	0.487	0.434	**0.381**	0.384	0.382
Seattle	0.458	0.489	0.436	0.409	**0.399**	**0.399**
Avg.	0.461	0.489	0.439	0.392	0.390	**0.382**

Table 4: Item-oriented – Comparison between values of *k*

		*i*VS-CNN			
Metric		LowConv	MidConv	HighConv	FC
Pointwise Accuracy	k = 8	0.561	0.607	0.600	0.605
(higher is better)	k = 16	0.563	0.610	0.612	0.620
Pairwise Accuracy	k = 8	0.590	0.655	0.644	0.666
(higher is better)	k = 16	0.592	0.655	0.660	0.678
MAE	k = 8	0.441	0.395	0.401	0.397
(lower is better)	k = 16	0.439	0.392	0.390	0.382

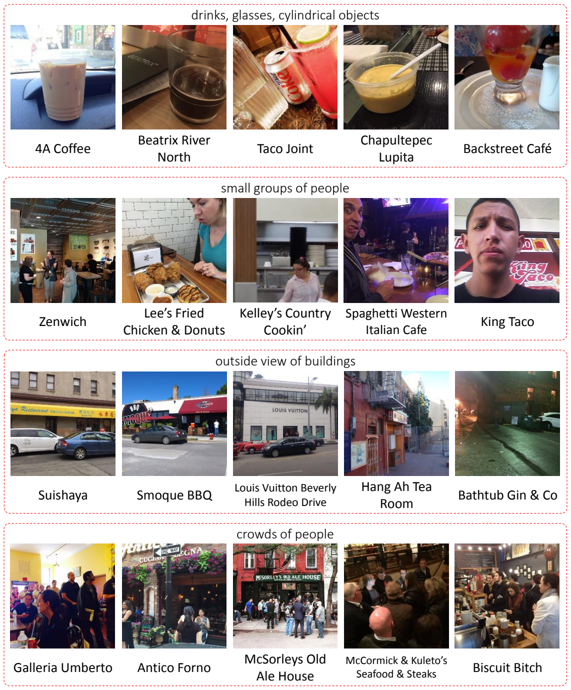

Figure 6: Item-oriented – Images from "contrarian" items in *i*VS-CNN that reverse the positive classification of VS-CNN.

Case Study. To get an intuition of the kind of review images that connote positive or negative sentiment visually, we illustrate several examples. First, we look at the images with the highest probability for positive class by the base model VS-CNN. These are images that are generally considered positive by most items.

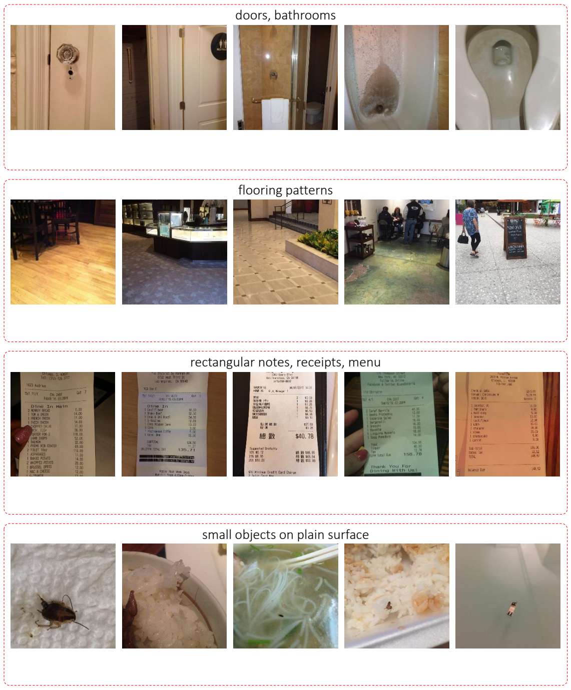
doors, bathrooms
flooring patterns
rectangular notes, receipts, menu
small objects on plain surface

Figure 7: Item-oriented – Most negative images from the base model VS-CNN.

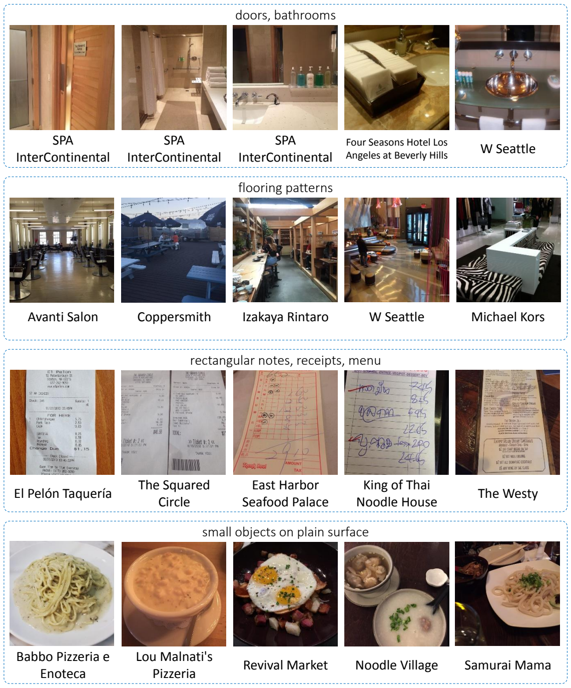

Figure 8: Item-oriented – Images from "contrarian" items in *i*VS-CNN that reverse negative classification of VS-CNN.

Figure 5 shows four clusters of images. The leftmost image in each cluster is one of the top-ranking images in terms of probability. The other images in the cluster are those that are its nearest neighbors (in terms of the feature representation at *fc7* layer). The first cluster is about drinks. The second is about one or two persons, in some cases celebrating something (cake and candle). The third shows outside views, probably in well-located restaurants with good views. The last shows a group of people taking a picture together.

To understand how item-orientation could have an effect, for each cluster of positive images in Figure 5, we identify items (businesses) whose models would reverse the positive classification of the cluster of images into negative classification. These are "contrarian" items that differ from the general population. Figure 6 showcases images from these contrarian items (names noted under each image), each cluster corresponding to a cluster in Figure 5. The first cluster is also about drinks, but construed negatively. The second cluster is also about people, but not in celebratory mood. The third shows outside views of restaurants, probably implying parking situations. The last shows crowds of people lining up.

In turn, Figure 7 shows images considered negative by the base model VS-CNN. As a contrast, Figure 8 showcases images from "contrarian" items that would have considered those same images positive. The first cluster is about toilet, and the second cluster is about flooring. Interestingly, those in Figure 8 show more "upscale" versions. Not surprisingly, the third cluster captures receipts, implying that some businesses may not deliver good value. The cluster also captures some menus (similar to some receipts) that for some businesses may be considered positive. The last captures small object on white surface, which unfortunately may be insect or bug in food (negative sentiment) in Figure 7 , but fortunately may be tiny condiments on dishes (positive sentiment) in Figure 8.

4.2 Experiments with User-oriented CNN

We now investigate the effect of user-orientation against the base model. From the *Yelp.com* crawl, we extract a balanced dataset via stratified random sampling so each user has the same number of positive and negative images. In contrast to the item-oriented iVS-CNN experiment, for user-oriented uVS-CNN, we maintain this balance for each user. It may not be possible to construct one dataset that maintain the balance for both users and items simultaneously. This dataset has 61,720 images involving 11,718 users and 8,133 businesses. On average, each user has 5.3 images from 3.1 businesses. We sample 80% for training and 20% for testing.

Quantitative Evaluation. Table 5 shows the comparison between the Naive Bayes classifier NB, the base model VS-CNN and the user-oriented *u*VS-CNN in terms of pointwise accuracy at $k = 16$. The base model's accuracy is 0.539 and a little bit lower than NB of 0.544. Similar observations as before can be made on the increasing accuracies that can be reached by factoring the user-orientation into higher levels of abstraction, from the low to mid to high-level convolutions and finally to the fully-connected layer with accuracy of 0.649. Compared to the item-oriented experiments in Table 1, the accuracy for the base model VS-CNN is now lower, while that of the user-oriented *u*VS-CNN is higher than *i*VS-CNN. This could imply that there are greater variations across users than across items (businesses), such that factoring users could pay off

Table 5: User-oriented – Pointwise Accuracy (higher is better)

City	NB	VS-CNN	*u*VS-CNN			
			LowConv	MidConv	HighConv	FC
Boston	0.537	0.546	0.570	0.610	**0.644**	**0.644**
Chicago	0.535	0.534	0.607	0.628	**0.646**	0.642
Houston	0.536	0.540	0.580	0.617	0.625	**0.629**
Los Angeles	0.550	0.540	0.594	0.639	0.651	**0.661**
New York	0.541	0.539	0.596	**0.657**	0.654	0.646
San Francisco	0.568	0.553	0.605	0.651	0.651	**0.668**
Seattle	0.528	0.516	0.617	0.627	0.623	**0.630**
Avg.	0.544	0.539	0.596	0.638	0.646	**0.649**

Table 6: User-oriented – Pairwise Accuracy (higher is better)

City	NB	VS-CNN	*u*VS-CNN			
			LowConv	MidConv	HighConv	FC
Boston	0.542	0.531	0.619	0.700	0.703	**0.727**
Chicago	0.554	0.539	0.634	0.690	0.696	**0.749**
Houston	0.569	0.562	0.613	0.662	**0.721**	0.708
Los Angeles	0.561	0.567	0.637	0.691	0.706	**0.751**
New York	0.556	0.550	0.647	0.706	0.716	**0.742**
San Francisco	0.604	0.589	0.654	0.676	0.711	**0.769**
Seattle	0.542	0.534	0.655	0.634	0.671	**0.720**
Avg.	0.562	0.556	0.639	0.686	0.706	**0.743**

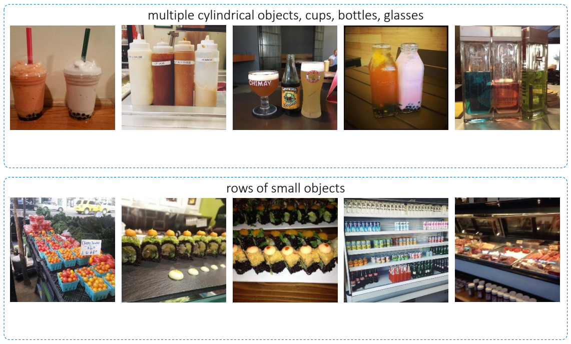

Figure 9: User-oriented – Most positive images from the base model VS-CNN.

more. This observation is also borne by the pairwise accuracy (see Table 6), which compares two images of the same user, and the MAE (see Table 7). The summary results in Table 8 show that *u*VS-CNN tends to perform better at $k = 16$ than at $k = 8$. Notably, for either setting of k, *u*VS-CNN still outperforms the base model VS-CNN.

Case Study. We illustrate another case study, but this time for user-orientation, much more briefly than before due to space limitation. Figure 9 shows two clusters of images considered positive by the base model VS-CNN. The first shows images of several cyclindrical objects, including sauces or drinks. The second shows rows of small objects, including fruits, sushi rolls, cakes, etc. In turn, Figure 10 shows images by "contrarian" users who would consider the images in Figure 9 to be negative, as they may be associated with those users' negative reviews. *u*VS-CNN manages to capture the peculiarities of some users in interpreting the image sentiments.

Table 7: User-oriented – MAE (lower is better)

City	NB	VS-CNN	*u*VS-CNN			
			LowConv	MidConv	HighConv	FC
Boston	0.464	0.496	0.432	0.393	0.357	**0.356**
Chicago	0.463	0.494	0.402	0.368	**0.353**	0.359
Houston	0.465	0.493	0.423	0.382	**0.374**	0.375
Los Angeles	0.451	0.493	0.412	0.362	0.351	**0.342**
New York	0.460	0.496	0.408	**0.346**	0.347	0.357
San Francisco	0.433	0.493	0.400	0.357	0.349	**0.335**
Seattle	0.471	0.497	0.395	0.377	0.378	**0.371**
Avg.	0.456	0.494	0.410	0.364	0.355	**0.353**

Table 8: User-oriented – Comparison between values of k

		*u*VS-CNN			
Metric		LowConv	MidConv	HighConv	FC
Pointwise Accuracy	k = 8	0.604	0.626	0.628	0.640
(higher is better)	k = 16	0.596	0.638	0.646	0.649
Pairwise Accuracy	k = 8	0.653	0.679	0.685	0.731
(higher is better)	k = 16	0.639	0.686	0.706	0.743
MAE	k = 8	0.398	0.375	0.373	0.362
(lower is better)	k = 16	0.410	0.364	0.355	0.353

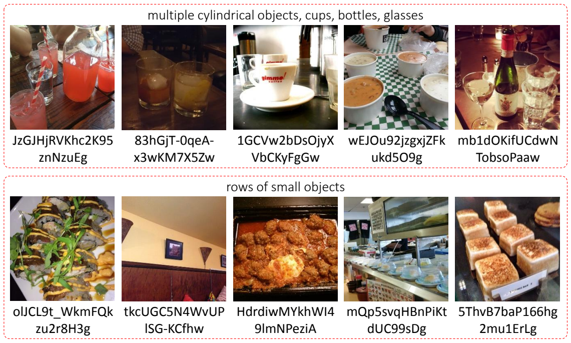

Figure 10: User-oriented – Images from "contrarian"users in *u*VS-CNN that reverse positive classification of VS-CNN.

5 CONCLUSION

We hypothesize that review images contain sentiment signals. Indeed the base model achieves higher accuracies than random. We further investigate the roles of item-orientation and user-orientation. Some image features may code for positive sentiment for some items, and yet code for negative sentiment for others. Experiments show that the item-oriented CNN achieves even higher accuracies, particularly when item-orientation is incorporated at higher levels of abstraction. Experiments for user-orientation yield similar results. As future work, we would analyse how review text could be used with review images for multi-modal sentiment analysis.

ACKNOWLEDGMENTS

This research is supported by the National Research Foundation, Prime Minister's Office, Singapore under its NRF Fellowship Programme (Award No. NRF-NRFF2016-07).

REFERENCES

[1] Marian Stewart Bartlett, Gwen Littlewort, Mark Frank, Claudia Lainscsek, Ian Fasel, and Javier Movellan. 2005. Recognizing facial expression: machine learning and application to spontaneous behavior. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), 20-26 June 2005, San Diego, CA, USA*, Vol. 2. IEEE, 568–573.

[2] Johan Bollen, Huina Mao, and Alberto Pepe. 2011. Modeling Public Mood and Emotion: Twitter Sentiment and Socio-Economic Phenomena. In *Proceedings of the Fifth International Conference on Weblogs and Social Media, Barcelona, Catalonia, Spain, July 17-21, 2011*. The AAAI Press.

[3] Damian Borth, Tao Chen, Rongrong Ji, and Shih-Fu Chang. 2013. SentiBank: large-scale ontology and classifiers for detecting sentiment and emotions in visual content. In *ACM Multimedia Conference, MM '13, Barcelona, Spain, October 21-25, 2013*. ACM, 459–460.

[4] Tao Chen, Damian Borth, Trevor Darrell, and Shih-Fu Chang. 2014. DeepSentiBank: Visual Sentiment Concept Classification with Deep Convolutional Neural Networks. *arXiv preprint arXiv:1410.8586* (2014).

[5] Yan-Ying Chen, Tao Chen, Winston H Hsu, Hong-Yuan Mark Liao, and Shih-Fu Chang. 2014. Predicting viewer affective comments based on image content in social media. In *International Conference on Multimedia Retrieval, ICMR '14, Glasgow, United Kingdom - April 01 - 04, 2014*. ACM, 233.

[6] Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010. Enhanced Sentiment Learning Using Twitter Hashtags and Smileys. In *COLING 2010, 23rd International Conference on Computational Linguistics, Posters Volume, 23-27 August 2010, Beijing, China*. Chinese Information Processing Society of China, 241–249.

[7] Wei Di, Neel Sundaresan, Robinson Piramuthu, and Anurag Bhardwaj. 2014. Is a picture really worth a thousand words?: - on the role of images in e-commerce. In *Seventh ACM International Conference on Web Search and Data Mining, WSDM 2014, New York, NY, USA, February 24-28, 2014*. ACM, 633–642.

[8] Anjan Goswami, Naren Chittar, and Chung H. Sung. 2011. A study on the impact of product images on user clicks for online shopping. In *Proceedings of the 20th International Conference on World Wide Web, WWW 2011, Hyderabad, India, March 28 - April 1, 2011 (Companion Volume)*. ACM, 45–46.

[9] Mitchell S Green. 2007. *Self-expression*. Oxford University Press.

[10] Ruining He, Chunbin Lin, Jianguo Wang, and Julian McAuley. 2016. Sherlock: Sparse Hierarchical Embeddings for Visually-Aware One-Class Collaborative Filtering. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*. IJCAI/AAAI Press, 3740–3746.

[11] Ruining He and Julian McAuley. 2016. VBPR: Visual Bayesian Personalized Ranking from Implicit Feedback. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA*. AAAI Press, 144–150.

[12] Xia Hu, Jiliang Tang, Huiji Gao, and Huan Liu. 2013. Unsupervised sentiment analysis with emotional signals. In *22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013*. International World Wide Web Conferences Steering Committee / ACM, 607–618.

[13] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross B. Girshick, Sergio Guadarrama, and Trevor Darrell. 2014. Caffe: Convolutional Architecture for Fast Feature Embedding. In *Proceedings of the ACM International Conference on Multimedia, MM '14, Orlando, FL, USA, November 03 - 07, 2014*. ACM, 675–678.

[14] Yannis Kalantidis, Lyndon Kennedy, and Li-Jia Li. 2013. Getting the look: clothing recognition and segmentation for automatic product suggestions in everyday photos. In *International Conference on Multimedia Retrieval, ICMR'13, Dallas, TX, USA, April 16-19, 2013*. ACM, 105–112.

[15] Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*. ACL, 1746–1751.

[16] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *IEEE Computer* 42, 8 (2009), 30–37.

[17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States*. 1106–1114.

[18] Yann LeCun, Bernhard E. Boser, John S. Denker, Donnie Henderson, R. E. Howard, Wayne E. Hubbard, and Lawrence D. Jackel. 1989. Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation* 1, 4 (1989), 541–551.

[19] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based Learning Applied to Document Recognition. *Proc. IEEE* 86, 11 (1998), 2278–2324.

[20] Yann LeCun, Koray Kavukcuoglu, and Clément Farabet. 2010. Convolutional networks and applications in vision. In *International Symposium on Circuits and Systems (ISCAS 2010), May 30 - June 2, 2010, Paris, France*. IEEE, 253–256.

[21] Guangxia Li, Steven C. H. Hoi, Kuiyu Chang, and Ramesh Jain. 2010. Microblogging Sentiment Detection by Collaborative Online Learning. In *ICDM 2010, The 10th IEEE International Conference on Data Mining, Sydney, Australia, 14-17 December 2010*. IEEE Computer Society, 893–898.

[22] Bo Pang, Lillian Lee, and others. 2008. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval* 2, 1–2 (2008), 1–135.

[23] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing-Volume 10*. Association for Computational Linguistics, 79–86.

[24] Stefan Siersdorfer, Enrico Minack, Fan Deng, and Jonathon S. Hare. 2010. Analyzing and predicting sentiment of images on the social web. In *Proceedings of the 18th International Conference on Multimedia 2010, Firenze, Italy, October 25-29, 2010*. ACM, 715–718.

[25] Andranik Tumasjan, Timm Oliver Sprenger, Philipp G. Sandner, and Isabell M. Welpe. 2010. Predicting Elections with Twitter: What 140 Characters Reveal about Political Sentiment. In *Proceedings of the Fourth International Conference on Weblogs and Social Media, ICWSM 2010, Washington, DC, USA, May 23-26, 2010*. The AAAI Press.

[26] Yilin Wang, Yuheng Hu, Subbarao Kambhampati, and Baoxin Li. 2015. Inferring Sentiment from Web Images with Joint Inference on Visual and Social Cues: A Regulated Matrix Factorization Approach. In *Proceedings of the Ninth International Conference on Web and Social Media, ICWSM 2015, University of Oxford, Oxford, UK, May 26-29, 2015*. AAAI Press, 473–482.

[27] Quanzeng You, Jiebo Luo, Hailin Jin, and Jianchao Yang. 2015. Robust Image Sentiment Analysis Using Progressively Trained and Domain Transferred Deep Networks. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA*. AAAI Press, 381–388.

[28] Quanzeng You, Jiebo Luo, Hailin Jin, and Jianchao Yang. 2016. Cross-modality consistent regression for joint visual-textual sentiment analysis of social multimedia. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining, San Francisco, CA, USA, February 22-25, 2016*. ACM, 13–22.

[29] Jianbo Yuan, Sean Mcdonough, Quanzeng You, and Jiebo Luo. 2013. Sentribute: image sentiment analysis from a mid-level perspective. In *Proceedings of the Second International Workshop on Issues of Sentiment Discovery and Opinion Mining, WISDOM 2013, Chicago, IL, USA, August 11, 2013*. ACM, 10:1–10:8.