User Conditional Hashtag Prediction for Images

Emily Denton*
New York University
denton@cs.nyu.edu

Jason Weston
Facebook AI Research
jase@fb.com

Manohar Paluri
Facebook AI Research
mano@fb.com

Lubomir Bourdev
Facebook AI Research
lubomir@fb.com

Rob Fergus
Facebook AI Research
robfergus@fb.com

ABSTRACT

Understanding the content of user's image posts is a particularly interesting problem in social networks and web settings. Current machine learning techniques focus mostly on curated training sets of image-label pairs, and perform image classification given the pixels within the image. In this work we instead leverage the wealth of information available from users: firstly, we employ user hashtags to capture the description of image content; and secondly, we make use of valuable contextual information about the user. We show how user metadata (age, gender, etc.) combined with image features derived from a convolutional neural network can be used to perform hashtag prediction. We explore two ways of combining these heterogeneous features into a learning framework: (i) simple concatenation; and (ii) a 3-way multiplicative gating, where the image model is conditioned on the user metadata. We apply these models to a large dataset of de-identified Facebook posts and demonstrate that modeling the user can significantly improve the tag prediction quality over current state-of-the-art methods.

Categories and Subject Descriptors

I.2.6 [**Learning**]: Connectionism and neural nets; I.5.1 [**Models**]: Neural nets; I.5.4 [**Applications**]: Computer vision

Keywords

Social media, user modeling, deep learning, hashtagging, large scale image annotation

1. INTRODUCTION

Hashtags (single words, abbreviations or word concatenations, prefixed by the # symbol) commonly accompany online image content, most notably on social media platforms. Far richer than conventional semantic labels, they provide an incredibly varied and nuanced method of describing images. Some hashtags describe precise labels (`#puppy`, `#craftbeer`). Others contain information about the feelings and intent of the user, e.g. reflecting happiness or sadness (`#awesome`, `#whyme`), or refer to some event in the person's past, present or future (`#happyhour`, `#babyshower`). There are also a wide variety of popular hashtags that convey abstract ideas, and are not necessarily tied to particular image content (`#nofilter`, `#sundayfunday`). Common hashtags that we reference in this paper are defined in Table 7. Hashtags capture the ever-changing distribution of user interests: new hashtags that have never occurred before are constantly being created in response to recent events, products or newly famous people, some of which can become popular very quickly. Finally, a given user, as well as having particular biases of the kind of images they will upload, also have biases of the kind of hashtags they choose to write. Figure 1 shows examples of hashtagged images uploaded to Facebook.

Modern systems for understanding web content make extensive use of machine learning methods for image recognition. In particular, deep learning methods such as convolutional networks have become very popular due to their

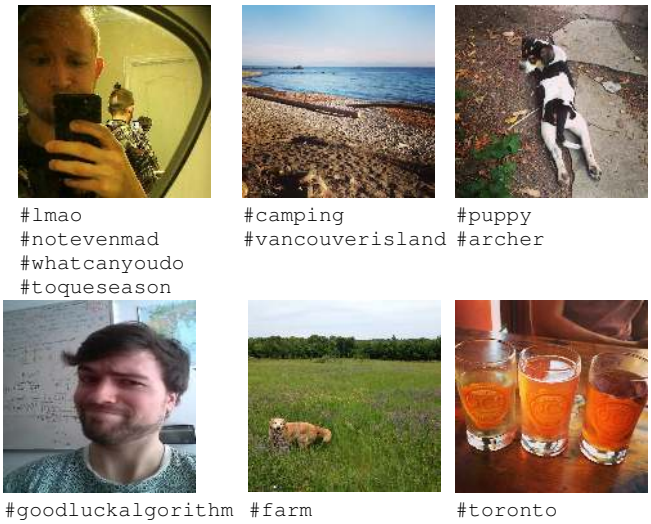


Table 1: Examples of hashtagged images. Images used with owners permission.

*Work done while at Facebook AI Research

KDD'15, August 10-13, 2015, Sydney, NSW, Australia.

DOI: http://dx.doi.org/10.1145/2783258.2788576.

impressive performance [7]. Training such models has typically depended on large sets of manually annotated data (e.g. Imagenet [1]), which is time-consuming and arduous to obtain. Further, such data ignores several aspects of image understanding that are of particular interest to web users: (i) their focus is on precise physical description so aspects such as sentiment are not addressed; (ii) the data distribution differs from online data and it is also unlikely to adapt quickly to changing user interests; and (iii) labels are independent of the users who originally authored the images or image posts (e.g. they are labeled by a Mechanical Turk worker).

In this work, we consider the vast amount of image content on the web where users have provided hashtags as a powerful, alternative training data source. In addition to generating very large amounts of labeled data compared to a manually curated set, we can also directly train on the actual data we wish to capture, rather than one whose distribution differs from user's interests. We thus define our training task as follows: we wish to predict the hashtags for a given *image* uploaded by a given *user*. The learning techniques we employ thus contain feature representations modeling both the image via pixels and the user in the form of metadata. Our hypothesis is that the combination of these sources provides useful information.

We consider several possible architectures based on embedding models. Embeddings are vector representations of images, metadata or text. Embedding models have been successfully used in a variety of contexts such as large scale image annotation [17], zero-shot learning of image categories [2] and hashtag prediction for text posts [18]. An embedding model maps inputs and hashtags into a common embedding space. For a given input, hashtags are ranked according to the dot product between hashtag embedding vectors and the image embedding vector. For the image features, a pre-trained convolutional neural network acts as a feature extractor, converting a raw input image into a concise image descriptor. We present three different methods of embedding inputs. The simplest method learns a linear mapping from image descriptors to embedding space. We then introduce two different methods of incorporating user metadata into the embedding process: (i) simple concatenation; and (ii) a 3-way multiplicative gating, where the image model is conditioned on the user metadata. In our experiments we incorporate the user's gender, age and city into the prediction. Our proposed methods are general, robust and scalable. As such, they can be combined with a variety of machine learning approaches and also deployed in large-scale real-world situations.

Image hashtag prediction has a variety of applications. For example, social media sites could use such a system to recommend hashtags to users as they upload image content. They can also be used for image search or for recommendation and ranking images based on content. Hashtags can also fulfill other roles such as disambiguating synonyms (e.g. jaguar `#car` vs jaguar `#bigcat`), or identifying entities (`#nyc`). In our experiments we apply our models to a large dataset of de-identified Facebook posts and demonstrate that modeling the image and user can significantly improve the tag prediction quality over current state-of-the-art methods.

2. RELATED WORK

As mentioned in Section 1, embedding models have been used in a variety of domains. Of particular relevance to our work is WSABIE, a supervised embedding approach that has been used for large scale image annotation and NLP tasks [17, 3]. Two of the models we present in Section 3 have a linear embedding function based on the WSABIE model and all our models are trained with the ranking loss presented in this work.

The most sophisticated of our proposed embedding models incorporates user information via a three-way tensor product. Multiplicative three-way connections have been used as a feature gating mechanism in restricted Boltzmann machines [15, 20, 9, 14]. More recently, this type of three-way connection has been used to condition neural language models on other modalities [6]. This work is similar to ours in that the tensor product is used to combine inputs from two different domains.

Previous work [18, 5] has addressed the problem of hashtag prediction for text posts. The work of [18] is most similar to our work since we both propose hashtag embedding models trained with a ranking loss. In addition to the obvious difference in domains, our work differs in that we introduce a novel method of incorporating user metadata into hashtag prediction.

Many works have incorporated user modeling into machine learning methods for various applications, for example by considering tensor factorisations for recommendation using music tags [13], web pages [10], using age and gender [4], location [21] and time [19]. However, we are not aware of similar works for our particular application in the image tagging domain.

3. HASHTAG EMBEDDING MODEL

We propose a hashtag embedding model that learns a joint d-dimensional embedding space for hashtags and image posts. Let $y \in \mathcal{Y} = \{1, \ldots, Y\}$ denote an index into a dictionary of possible hashtags. Let $x \in \mathbb{R}^n$ be an image descriptor. The model is of the form:

$$f(x, y) = \Phi_I(x)^\top \Phi_H(y)$$

where $\Phi_I(x) : \mathbb{R}^n \to \mathbb{R}^d$ is an image embedding function and $\Phi_H(y) : \mathcal{Y} \to \mathbb{R}^d$ is a hashtag embedding function.

For a given image, x, possible hashtags, y, are ranked according to $f(x, y)$. We optimize precision at the top of the ranked list using the WARP loss as described in section 3.4.

The hashtag embedding function is given by

$$\Phi_H(y) = V_y$$

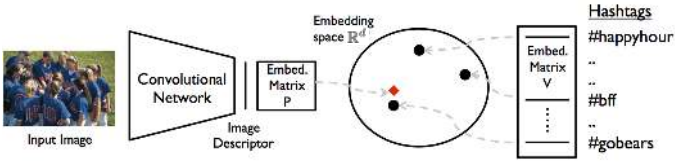


Figure 1: Illustration of bilinear image embedding model.

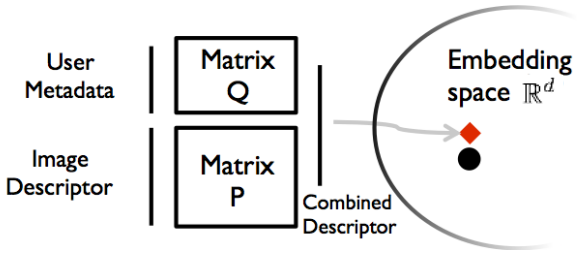


Figure 2: How the user metadata is combined with the image features in the user-biased model

.

where $V \in \mathbb{R}^{Y \times d}$ is the hashtag embedding matrix and V_i indexes the i^{th} row of V. Following [17], we constrain the hashtag embedding matrix such that

$$\|V_i\|_2 \leq 1, i = 1, \ldots, Y$$

This regularization technique helps prevent the model from overfitting.

The image embedding function takes an image descriptor as input and produces a d-dimensional embedding vector. We use a pre-trained convolutional neural network [8] as a feature extractor to obtain the image descriptor from the raw image post. The convolutional network was trained separately on ∼1 million Facebook images. The network was trained to classify 1000 different concepts which span multiple categories such as Object, Scenes, Actions, Food, Animals, etc. The image descriptors of dimensionality $n = 4096$ are extracted from the final fully connected layer of the network, which has an architecture similar to that of [7].

We propose three different methods of embedding image descriptors. The simplest model does not use any user information and embeds an image descriptor via a linear mapping. We also propose two different methods of incorporating user metadata into the embedding process. The user-biased model introduces a user-dependent additive bias into the image embedding function and the user-multiplicative model has a user descriptor gate, via a multiplication operation, the image embedding function. We now describe each of these models in detail.

3.1 Bilinear Model

The bilinear embedding model [17] does not incorporate any user information into the image embedding function. The image embedding function is a linear map of the form

$$\Phi_I(x) = Px$$

where $P \in \mathbb{R}^{d \times n}$ is the image embedding matrix. As with the hashtag embedding matrix, we constrain the norm of P:

$$\|P_i\|_2 \leq 1, i = 1, \ldots n$$

to help prevent overfitting. The bilinear embedding model is illustrated in Figure 1.

3.2 User-biased Bilinear Model

Assume now that we have information associated with the users responsible for each image post. Let $u \in \mathbb{R}^m$ denote a vector representation of a user. The user-biased model gives a simple method for leveraging user information by adding a user dependent bias term to the image embedding function. In particular, the user-biased image embedding, $\Phi_{I+U}(x, u) : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^d$, is defined as

$$\Phi_{I+U}(x, u) = Px + Qu$$

where $Q \in \mathbb{R}^{d \times m}$ is the user embedding matrix. As in the bilinear model, we regularize the model by constraining the norms of P and Q:

$$\|P_i\|_2 \leq 1, i = 1, \ldots, n$$
$$\|Q_i\|_2 \leq 1, i = 1, \ldots, m$$

The user-biased embedding model is illustrated in Figure 2.

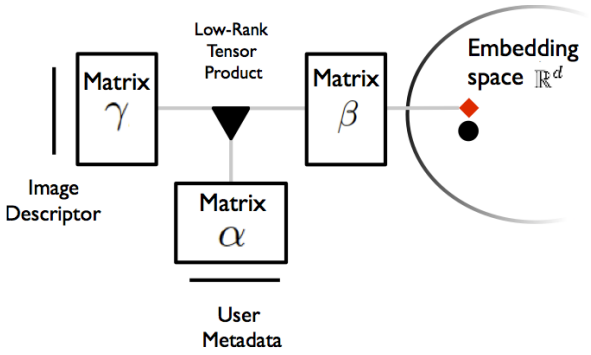


Figure 3: How the user metadata is combined with the image features in the user-multiplicative model.

3.3 User-multiplicative Tensor Model

The user-multiplicative model provides a more sophisticated method of incorporating user data into the image embedding function by letting the user feature vector, u, gate the image embedding. This allows every user vector to define a different image embedding matrix. The matrix multiplication from the bilinear and user-biased models is replaced with a product between an image descriptor, a user descriptor and an image embedding tensor.

Formally, let $Z \in \mathbb{R}^{m \times d \times n}$ denote the image embedding tensor. Then, for a given user descriptor, u, the user-multiplicative image embedding matrix is

$$P^u = \sum_{i=1}^{m} u_i Z^{(i)}$$

where u_i is the i^{th} component of u and $Z^{(i)} \in \mathbb{R}^{d \times n}$ is a slice of the image embedding tensor specified by the i^{th} index along the first dimension. The user-multiplicative image embedding function is then

$$\Phi_{I \times U}(x, u) = P^u x$$

. The tensor Z has mdn parameters which is impractical to train. We therefore constrain Z to have the following factorized form:

$$Z = \sum_{k=1}^{K} \alpha_k \otimes \beta_k \otimes \gamma_k$$

where $\alpha_k \in \mathbb{R}^m$, $\beta_k \in \mathbb{R}^d$, $\gamma_k \in \mathbb{R}^n$ where $\otimes$ is the tensor outer product operation. K denotes the number of factors and specifies the rank of Z. It is useful to represent the user factors, embedding factors and image factors in matrix form

as $\alpha \in \mathbb{R}^{m \times K}$, $\beta \in \mathbb{R}^{d \times K}$ and $\gamma \in \mathbb{R}^{n \times K}$. In this constrained form, Z has $K(m+d+n)$ free parameters. Section A of the appendix explains the 3-way multiplicative model in further detail, including the gradient computations performed during training.

As it is unclear how to choose an appropriate vector representation for user information, we learn one instead. This is done using a neural network which converts the user metadata to an appropriate vector prior to the α matrix in the 3-way connection. The full picture of this model is shown in Figure 3. In our experiments we use a single layer neural network with 24 hidden units (i.e. the user descriptor acted upon by the 3-way connection is 24-dimensional) with Rectified Linear Units [11]. The weights of this network, along with the 3-way factors, are learned jointly. Adding this extra layer makes the model more powerful and also provides a way of extracting concise user vectors from the hidden layer activations, which could potentially be transferred to other tasks, beyond the one considered in this paper.

3.4 Training Algorithm

We train our models by minimizing the weighted approximate-rank pairwise (WARP) loss as described in [17]. The WARP loss approximately optimizes precision at k using a negative sampling technique. This type of loss is ideal for our task since it easily scales to large hashtag vocabularies.

The algorithm proceeds as follows: At every iteration, we sample a positive example, (x, y^+). Up to 1000 negative hashtags, y^-, are then sampled in an attempt to find one such that

$$f(x, y^-) > f(x, y^+) - m$$

where m specifies the margin. We then take a gradient step to minimize

$$|m - f(x, y^+) + f(x, y^-)|_+$$

Following [18], we set $m = 0.1$ in all our experiments.

We minimize the loss with parallel stochastic gradient descent with the hogwild algorithm [12]. Weights were initialized from a zero-mean Gaussian distribution with standard deviation 0.001. The learning rate was initialized to 0.01 and was manually reduced by a factor of 10 when performance stopped improving on a validation set. We used a momentum constant of 0.9. We found training of the 3-way connection was significantly improved by normalizing the image descriptors to have L2 norm of 1 and so this step was performed for all models.

4. EXPERIMENTS

4.1 Dataset

We evaluate our image embedding models on a large Facebook dataset, consisting of 20 million public images uploaded by ∼10.4 million de-identified users. The images were collected at random from uploads over a period of several days in October 2014. The images were selected to have at least one hashtag per image, but often had multiple hashtags. In total, the dataset contained ∼4.6 million distinct hashtags. The mean number of hashtags per image was 2.7, with a standard deviation of 3.1. A large fraction of hashtags describe the content of the image, with many synonyms used (e.g. `#cat`, `#catsofinstagram`, `#kitten`). Others describe the sentiment of the user, being only loosely related to the image content (e.g. `#love`, `#happy`, `#blessed`). Another group convey abstract ideas, generally unrelated to the image content (`#nofilter`, `#fundaysunday`, `#latergram`).

Meta data	Possible values
Age	13 − 114
Gender	Male, Female, Unknown
Home City	GPS coordinates
Country	United States, Canada, Great Britain, Australia, New Zealand

Table 2: Summary of user data

Their distribution, shown in Figure 4, is far from uniform: the top 10 hashtags account for 4% of the total; the top 100 for 11%, and 4 million of them appear less than 10 times throughout the whole dataset. Given the difficulties of predicting very infrequent hashtags, we limit our vocabulary to the top 10k most frequent hashtags. In our experiments, we use two different versions of the dataset: *natural* and *balanced*. In the former, we directly use the natural distribution of the 10k most prevalent hashtags (shown in blue in Figure 4). However, the uneven distribution means that the few common hashtags will dominate any error measure, and make it hard to predict infrequent hashtags. Furthermore, many of the most common hashtags (e.g. `#wcw`, `#tbt`, `#mcm`) have little to do with image content, so distract the models from predicting image-relevant hashtags. To remedy this, we create a *balanced* version of the dataset, where the 500 most common hashtags are downsampled to have the same frequency as the 501st. The resulting distribution (shown in red in Figure 4), is much closer to uniform and thus ameliorates both issues. Note that the images and users are the same for both versions of the dataset; only the hashtag distribution changes.

Each de-identified user had 4 pieces of metadata: age, gender, country and city. The metadata is summarized in

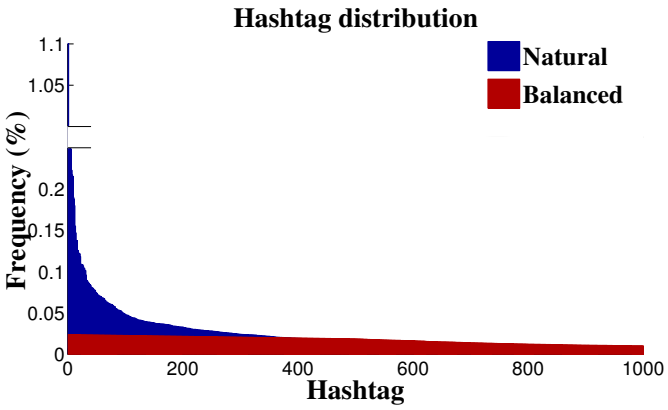


Figure 4: Natural (blue) and balanced (red) hashtag distributions used in our experiments. Note the break in the y-axis.

Method	d	K	P@1	R@10	A@10
Freq. baseline	-	-	3.04%	5.63%	9.45%
Bilinear	64	-	7.37%	11.71%	18.69%
Bilinear	128	-	7.37%	11.69%	18.44%
Bilinear	256	-	6.75%	10.84%	17.25%
Bilinear	512	-	6.50%	10.83%	17.17%
User-biased	64	-	9.02%	13.63%	21.88%
User-biased	128	-	9.00%	13.67%	21.83%
User-biased	256	-	8.48%	13.03%	20.96%
User-biased	512	-	7.98%	12.51%	20.05%
3-way mult.	64	50	8.95%	13.66%	21.82%
3-way mult.	64	100	9.03%	13.81%	22.04%
3-way mult.	64	200	8.96%	13.81%	22.05%
3-way mult.	64	300	9.00%	13.74%	21.96%
3-way mult.	64	400	8.96%	13.65%	21.82%

Table 3: Prediction results for models trained on data with a natural hashtag distribution.

Method	d	K	P@1	R@10	A@10
Freq. baseline	-	-	0.07%	0.36%	0.64%
Bilinear	64	-	3.75%	6.04%	9.81%
Bilinear	128	-	3.73%	5.81%	9.48%
Bilinear	256	-	3.54%	5.57%	9.06%
Bilinear	512	-	3.26%	5.10%	8.38%
User-biased	64	-	3.85%	6.22%	10.34%
User-biased	128	-	3.78%	6.13%	10.06%
User-biased	256	-	3.70%	5.85%	9.50%
User-biased	512	-	3.49%	5.78%	9.64%
3-way mult.	64	50	4.29%	6.94%	11.26%
3-way mult.	64	100	4.25%	6.80%	10.96%
3-way mult.	64	200	4.65%	7.27%	11.77%
3-way mult.	64	300	4.34%	7.02%	11.37%
3-way mult.	64	400	4.25%	7.04%	11.46%

Table 4: Prediction results on the balanced hashtag distribution.

Table 2. We restricted ourselves to images posted by users from five populous English speaking countries to ensure that the hashtags had a common underlying language. Imprecise location information for each de-identified user took the form of the GPS location of their home town/city[1]. The dataset was split into disjoint training and test sets, with each user only appearing in one of them (i.e. no overlap). The test set was 100k images in size, with the remainder used for training.

We construct a 10-dimensional user descriptor based on the 4 pieces of metadata. One dimension represents age as a continuous value, scaled to have value in [0, 1]. Another dimension represents gender, recorded as -1 or 1 for females and males respectively. An unknown gender is represented by 0. The country of origin is encoded as a 1-hot 5d vector for the five different countries. Finally, GPS coordinates are converted to 3-dimensional Cartesian coordinates and scaled to [0,1] range.

4.2 Results

We evaluate the embedding models described in Section 3 on both the natural and balanced versions of our dataset. For all three models, we vary d, the dimensionality of the embedding space to understand its effect on performance. For the 3-way multiplicative model, we also vary the rank K of the tensor factorization, which controls the number of parameters in the embedding function $\Phi_{I\times U}$. We include a simple baseline that ignores the test image and ranks hashtags according to their natural distribution in the training set.

The models are evaluated with three different metrics: precision, recall and accuracy. These are defined as follows: let $\text{Rank}(x, u, k)$ denote the set of top k ranked hashtags by the model for example (x, u) and let $\text{GroundTruth}(x, u)$ denote the set of hashtags actually tagged by the user u for the image x, then:

[1]I.e. the GPS coordinates of the town/city center associated with the user's profile, not the home address of the users, or where the photo was taken.

Precision:

$$\text{P@k} = \frac{1}{N}\sum_{i=1}^{N}\frac{|\text{Rank}(x_i, u_i, k) \cap \text{GroundTruth}(x_i, u_i)|}{|\text{Rank}(x_i, u_i, k)|}$$

Where N is the number of test examples. We show P@1 (i.e. what fraction of the time did the top ranked hashtag match one of the ground truth hashtags for the test image).

Recall:

$$\text{R@k} = \frac{1}{N}\sum_{i=1}^{N}\frac{|\text{Rank}(x_i, u_i, k) \cap \text{GroundTruth}(x_i, u_i)|}{|\text{GroundTruth}(x_i, u_i)|}$$

We show R@10 which gives a measure of the fraction of the relevant hashtags for each test image are ranked in the top 10. Note that if the image has more than 10 ground truth hashtags then this fraction will always be less than 1.

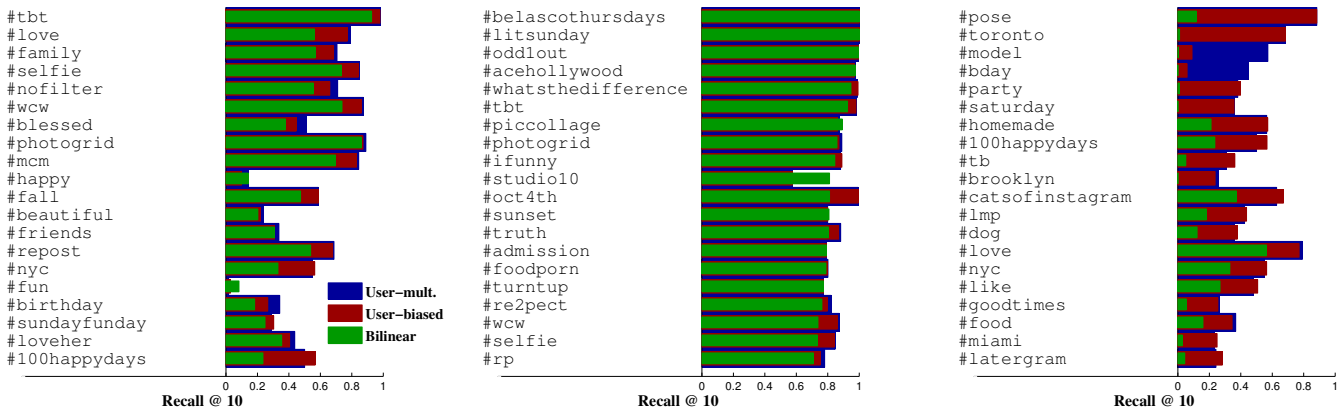
Accuracy:

$$\text{A@k} = \sum_{i=1}^{N}\frac{\text{I}[\text{Rank}(x_i, u_i, k) \cap \text{GroundTruth}(x_i, u_i) \neq \emptyset]}{N}$$

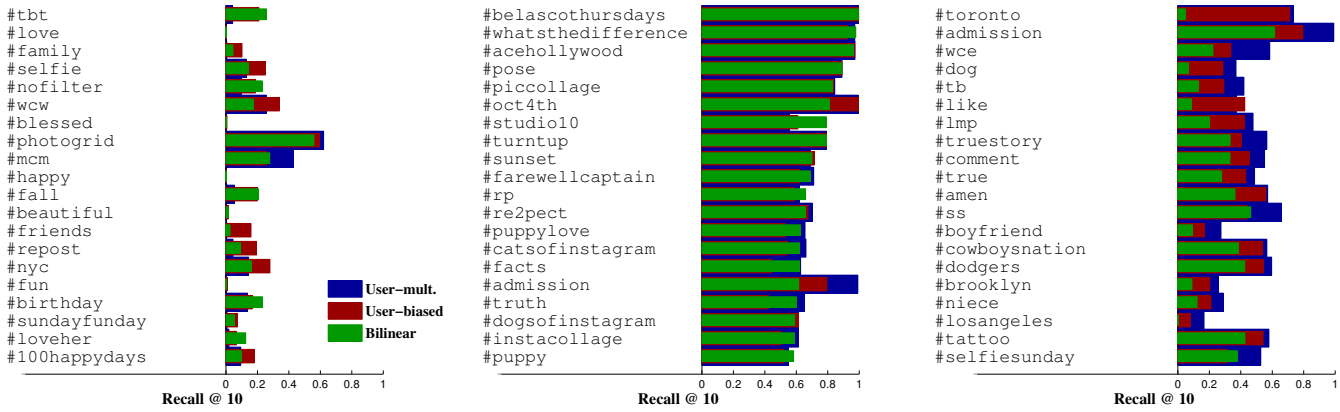
where $\text{I}[\cdot]$ is an indicator function. This metric is useful since it is largely indifferent to the number of ground truth hashtags. We show the accuracy at 10 (A@10) which gives a measure of how often at least one of the ground truth hashtags appears in the top 10 ranked hashtags.

Table 3 shows test results for models on the natural version of the dataset. Given that each image has 10k possible hashtags, the absolute values for all three metrics are low but, as the examples in Table 5 show, the quality of predicted hashtags is nevertheless reasonable. The predicted hashtags convey relevant image content (`#kayaking`, `#glasses`), social cues (`#reunion`, `#party`) and sentiment (`#goodtimes`, `#love`). The results in the table show: (i) all models clearly beating the frequency baseline; (ii) both models that incorporate user information significantly outperform the bilinear model; (iii) the user-multiplicative model performs comparably to the user-biased model; (iv) high values of the embedding dimension d hurt performance, likely due to overfitting and (v) the tensor rank K had little effort on performance.

Table 4 shows results from training on the balanced version of the dataset. The flatter hashtag distribution has higher entropy, relative to the natural version, making the



(a) Models trained on natural hashtag distribution.



(b) Models trained on balanced hashtag distribution.

Figure 6: Recall at 10 for different subsets of hashtags. Left: The 20 most frequent hashtags are shown. Middle: Top 20 hashtags which have highest recall for the bilinear model. Right: Top 20 hashtags which the difference in performance between the bilinear model and the multiplicative model is greatest.

prediction task harder as evidenced by the significant reduction in frequency baseline performance. In these experiments, the user-biased approach gives slight gains over bilinear. But the user-multiplicative model significantly outperforms both, showing it is able to make effective use of the user metadata. Examples of the predicted hashtags from this model are shown in Figure 5. The predictions can be seen to be far more varied and interesting than those from the same model trained on the natural version of the dataset. For example, very common hashtags such as `#tbt`, `#wcw`, `#mcm` are ranked in the top 10 for almost every photo by the model trained on the natural hashtag distribution. The model trained on the balanced distribution produces less frequent hashtags that are still relevant to the image.

Tables 3 and 4 show precision and recall values for images, averaged over all examples in the test set. Since some hashtags are predicted more accurately than others, we also show precision and recall values for individual hashtags. The average precision and recall at k for a particular hashtag, y, is defined as:

$$\text{R@k}(y) = \frac{\sum_{i=1}^{N} |\text{Rank}(x_i, u_i, k) \cap \{y\}|}{\sum_{i=1}^{N} |\text{GroundTruth}(x_i, u_i) \cap \{y\}|}$$

$$\text{P@k}(y) = \frac{\sum_{i=1}^{N} |\text{Rank}(x_i, u_i, k) \cap \{y\}|}{\sum_{i=1}^{N} |\text{Rank}(x_i, u_i, k)|}$$

Figure 5a plots precision-recall curves for a subset of hashtags for the three models, trained on the natural distribution. Some hashtags are better predicted by the models that leverage user information. For example, cities (`#nyc`, `#london`, etc.) see a large boost in performance when user information is incorporated into the model. Other hashtags that are not tied to specific image content but used predominately by certain demographics also see gains from the incorporation of user information. Hashtags such as `#100happydays`, `#wcw` and `#mcm` likely fall into this category. There are also a multitude of hashtags for which the addition of user information makes negligible difference, such as `#nature` or `#sunset`. These hashtags likely refer to specific image content, thus the addition of user information adds little signal.

Age	Females	Males
13-17	#mcm	#like
	#bestfriend	#lmp
	#love	#throwback
	#lovehim	#squad
	#mce	#wce
	#latepost	#throwback-
	#bestfriends	#thursday
	#boyfriend	#family
	#loveher	#workflow
	#loveyou	#selfie
		#wcm
43-47	#100happydays	#photoshop-
	#mcm	#express
	#love	#wcw
	#sisters	#goodtimes
	#cousins	#prouddad
	#lovehim	#throwback-
	#latergram	#thursday
	#loveher	#selfie
	#bff	#salute
	#youcampperfect	#blessed
		#zijasummit14
		#familyfirst

Gender	15-17 years old	43-47 years old
Female	#wcw	#100happydays
	#mcm	#blessed
	#bestfriend	#goodtimes
	#tb	#family
	#ss	#love
	#bestfriends	#photogrid
	#throwback	#latergram
	#latepost	#cousins
	#like	#sundayfunday
	#selfiesunday	#friends
Male	#wcw	#goodtimes
	#like	#blessed
	#throwback	#love
	#squad	#family
	#tb	#photoshop-
	#lmp	#express
	#mcm	#photogrid
	#ss	#sundayfunday
	#wce	#friends
	#selfiesunday	#zijasummit14
		#prouddad

Sydney	Toronto
#melbourne	#toronto
#sydney	#tbt
#australia	#canada
#spring	#vancouver
#beach	#fall
#grandfinal	#throwback
#sunshine	#blessed
#sun	#ilovethiscity
#nz	#vancity
#newzealand	#vscocam
#bali	#tb
#happy	#cntower
#nofilter	#goodmorning
#wellington	#hoco
#springbreak	#montreal
#bondi	#wcw
#afl	#tdot
#thailand	#lateupload
#stkilda	#downtown
#city	#beautiful

Table 6: An exploration of our multiplicative model using three different probe images. **Left:** Hashtags most affected by gender, for two different age groups. Left and right columns show hashtags predicted most frequently for female users relative to male users, and vice-versa. **Middle:** Hashtags most affected by age, for female and male users. Left and right columns show hashtags predicted most frequently for younger users relative to older users, and vice-versa. **Right:** Hashtags most affected by location. Left and right columns show hashtags that are predicted most frequently for Sydney users relative to Toronto users, and vice-versa.

Figure 5b plots precision-recall curves of a subset of hashtags for models trained on the balanced distribution. In this case we see the user-multiplicative model performs much better than the bilinear and user-biased models on most hashtags.

Figure 6a and Figure 6b show Recall@10 for several hashtags from the three models trained on the natural and balanced distributions respectively. In the leftmost plot we see the recall values for the 20 most frequent hashtags. For these very frequent hashtags, the models that incorporate user information perform comparably to the bilinear model. By comparing the leftmost plots in Figure 6a and Figure 6b we also see that the most frequent hashtags have much higher recall values for models trained on the natural hashtag distribution than for the models trained on the balanced distribution. The middle plot shows the top 20 hashtags for which the bilinear model has the highest recall, i.e. the hashtags that are predicted well without the aid of additional user information. Many of the hashtags with highest recall (`#belascothursdays`, `#litsunday`, `#acehollywood`) correspond to large event posters. Many different users post the event image and hashtag the image with the event name. Since the images and hashtags are nearly identical, the model is able to achieve nearly perfect recall. The right-hand plot shows the top 20 hashtags for which the difference in performance between the bilinear model and the multiplicative model is greatest. In other words, these figures show the hashtags whose accurate prediction relies most heavily on user information. We see that for these hashtags, the user-biased model and user-multiplicative models perform comparably when trained on the natural distribution. However, when trained on the balanced distribution, the user-multiplicative model tends to outperform both the bilinear and user-biased model.

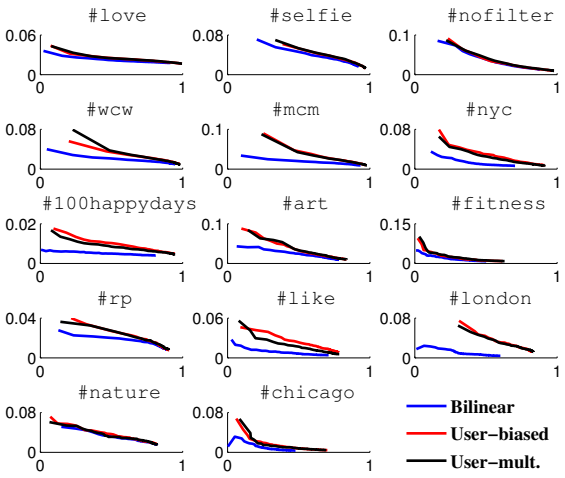
4.3 Understanding User Information

There are many types of images for which certain hashtags tend to be used more frequently by specific demographics. For example, given a cityscape image, hashtags such as `#toronto` and `#canada` are more likely to be used by individuals living in Toronto. Table 6 explores the user-multiplicative model to see which hashtags are most affected by gender, age and location. Taking gender (left column) as an example, this can be performed by the following procedure: (i) given an initial *probe* image we find a set of close neighbors[2] in the test set using their d-dimensional image descriptors. (ii) from this nearest-neighbor set we pick two different subsets, one for for males and another for females, whose age is constrained to be 13-17 (in the top row; age

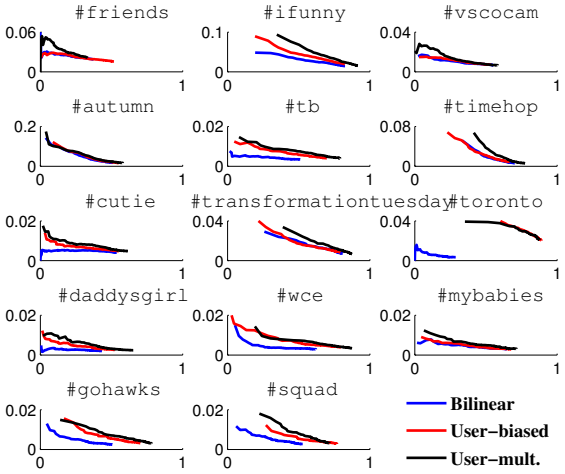
[2]We average over a large set of similar images to ensure the results are not specific to an individual image.

Input image	Top 10 ranked hashtags: Natural distribution	Top 10 ranked hashtags: Balanced distribution
	#tbt #squad #family #friends #wcw #homecoming #goodtimes #repost #throwback #party	#reunion #goodtimes #friends #reunited #drinks #squad #greatnight #party #drunk #fridaynight
	#nofilter #fall #autumn #beautiful #nature #vscocam #toronto #100happydays #sunshine #home	#nature #river #peaceful #lake #sunny #sunshine #trees #autumn #sun #beautifulday
	#selfie #nofilter #tbt #love #wcw #100happydays #mcm #cute #beautiful #happy	#selfie #glasses #nomakeup #haircut #smiles #sundayselfie #cutie #nofilter #selfies #photoshop- #express
	#nofilter #kingfire #sunset #fall #tbt #beautiful #blessed #nofilter- #needed #vscocam #goodmorning	#nofilter- #needed #blueangels #nofilter #sky #homesweethome #godscountry #farmlife #sunrise #countrylife #clouds
	#wcw #tbt #selfie #folsom #love #beyond- #wonderland #sundayfunday #family #beach #mcm	#beachday #beach #poolside #beachlife #funinthesun #poolparty #besties #beachtime #pooltime #sandiego
	#tbt #kayaking #nofilter #sundayfunday #mcm #wcw #family #familytime #latepost #love	#kayaking #pool #poolside #fishing #funinthesun #poolparty #pooltime #boating #boatlife #lakelife

Table 5: Top ranked hashtags from user-multiplicative model for sample test images. Left column shows hashtags predicted by model trained on the natural hashtag distribution. Right column shows hashtags from model trained on the balanced hashtag distribution. Photos used with owner permission.



(a) Models trained on the natural hashtag distribution.



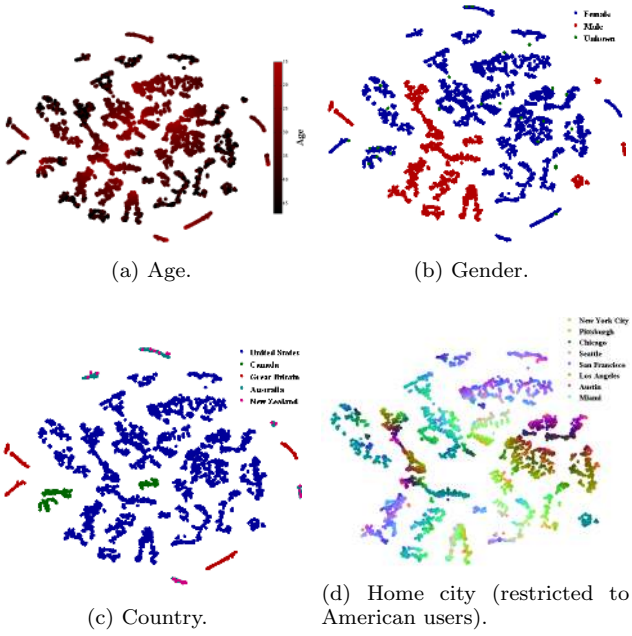
(b) Models trained on the balanced hashtag distribution.

Figure 5: Precision-recall curves of a subset of hashtags. Recall is plotted on the x-axis, precision on the y-axis.

43-47 for the bottom row). (iii) for each image in these two subsets we compute their embedding (using their associated metadata) and find the 10 closest hashtag embeddings. (iv) we compute aggregate counts of hashtags for the male and female subsets and display the hashtags where the *difference* is greatest between the two. The middle and right columns of the table also show how age (holding gender constant) and location affect the hashtag distribution, respectively.

In many cases, the hashtags strongly associated with a particular user attribute make intuitive sense. For example, the hashtags `#mcm`, `#lovehim` and `#boyfriend` are more frequently predicted for young female users than for young male users. The hashtag `#prouddad` is more frequently predicted for older male users than for female users of the same age. We also see that hashtags like `#beach` are predicted more often for Australian users than for those living in Toronto, whereas hashtags like `#cntower` are predicted more for Torontonians.

Figure 7 visualizes the learned user representation in our multiplicative model for 10k users in the test set. The 24-d user descriptors are mapped down to the 2D plane using the



(a) Age. (b) Gender.

(c) Country. (d) Home city (restricted to American users).

Figure 7: t-SNE visualization of user embeddings for 10K different users. Each plot is color coded by a different dimension of the user meta data.

t-SNE algorithm [16]. Four different color codings are used to show different user attributes. Clear structure is apparent in many cases, for example, the radius from the origin seems to correspond to increasing age.

5. CONCLUSIONS

We introduced a set of embedding models that predict highly diverse and relevant hashtags for real-world Facebook images. The simplest of these shows how image features derived from a convolutional neural network can be used to perform image hashtag prediction. We then showed how user metadata could be combined with image features to improve image hashtag prediction. The addition of user information gave a significant performance boost, particularly when incorporated in a multiplicative fashion.

Particular care has to be taken when working on real world datasets rather than curated ones. We addressed the highly skewed hashtag distribution observed in our dataset by downsampling the more frequent hashtags. We show that this technique produces more varied hashtag predictions.

Our models produce hashtags that capture many subtle social and sentiment of the images, and are far richer than the precise semantic descriptions output by many current recognition models. The resulting system is highly scalable and could be used in a number of applications such as automated hashtag suggestion, image search or for recommendation and ranking images based on content.

6. REFERENCES

[1] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Conference on Computer Vision and Pattern Recognition*, 2009.

[2] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov. Devise: A deep visual-semantic embedding model. In *Proc. of Advances in Neural Information Processing Systems*, 2013.

[3] M. Hermann, D. Das, J. Weston, and K. Ganchev. Semantic frame identification with distributed word representations. In *Proc. of ACL*, 2014.

[4] A. Karatzoglou, X. Amatriain, L. Baltrunas, and N. Oliver. Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering. RecSys '10, pages 79–86, 2010.

[5] E. Khabiri, J. Caverlee, and K. Kamath. Predicting semantic annotations on the real-time web. In *Proc. of 23rd ACM conference on Hypertext and social media*, pages 219–228, 2012.

[6] R. Kiros, R. Salakhutdinov, and R. Zemel. Multimodal neural language models. In *Proc. of The 30th International Conference on Machine Learning*, 2014.

[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proc. of Advances in Neural Information Processing Systems 25*, 2012.

[8] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, Nov 1998.

[9] R. Memisevic and G. E. Hinton. Learning to represent spatial transformations with factored higher-order boltzmann machines. *Neural Computation*, 22(6):1473–1492, 2010.

[10] A. Menon, K. Chitrapura, S. Garg, D. Agarwal, and N. Kota. Response prediction using collaborative filtering with hierarchies and side-information. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 141–149. ACM, 2011.

[11] V. Nair and G. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proc. of the 27th International Conference on Machine Learning*, 2010.

[12] F. Niu, B. Recht, C. Re, and S. J. Wright. Hogwild!: A lock-free approach to parallelizing stocastic gradient descent. In *Proc. of Advances in Neural Information Processing Systems 25*, 2011.

[13] S. Rendle and L. Schmidt-Thieme. Pairwise interaction tensor factorization for personalized tag recommendation. In *WSDM*, pages 81–90. ACM, 2010.

[14] K. Sohn, G. Zhou, C. Lee, and H. Lee. Learning and selecting features jointly with point-wise gated Boltzmann machines. In *Proc. of The 30th International Conference on Machine Learning*, pages 217–225, 2013.

[15] G. Taylor and G. E. Hinton. Factored conditional restricted boltzmann machines for modeling motion style. In *Proc. of the 26th International Conference on Machine Learning*, 2009.

[16] L. van der Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.

[17] J. Weston, S. Bengio, and N. Usunier. Wsabie: Scaling up to large vocabulary image annotation. In *Proc. of*

the 22nd international joint conference on Artificial Intelligence, 2011.

[18] J. Weston, S. Chopra, and K. Adams. #tagspace: Semantic embeddings from hashtags. In *Conference on Empirical Methods in Natural Language Processing*, pages 1822–1827. Association for Computational Linguistics, 2014.

[19] L. Xiong, X. Chen, T. Huang, J. Schneider, and J. Carbonell. Temporal collaborative filtering with bayesian probabilistic tensor factorization. In *Proceedings of SIAM Data Mining*, volume 2010, 2010.

[20] M. Zeiler, G. Taylor, L. Sigal, I. Matthews, and R. Fergus. Input-output temporal restricted boltzmann machines for facial expression transfer. In *Proc. of Advances in Neural Information Processing Systems 24*, 2011.

[21] V. Zheng, B. Cao, Y. Zheng, X. Xie, and Q. Yang. Collaborative filtering meets mobile recommendation: A user-centered approach. In *Proceedings of the 24rd AAAI Conference on Artificial Intelligence*, 2010.

APPENDIX

A. FACTORED 3-WAY GRADIENT COMPUTATIONS

The user-multiplicative model gates the image embedding matrix by a user descriptor, u. The user conditioned image embedding matrix is

$$P^u = \sum_{i=1}^{m} u_i Z^{(i)}$$

where $Z^{(1)}, \ldots, Z^{(m)}$ are $d \times n$ matrices and u_i indexes the i^{th} component of the user descriptor. Combining each of the $Z^{(i)}$'s into a tensor gives the image embedding tensor $Z \in \mathbb{R}^{m \times d \times n}$. To facilitate learning, Z is constrained to be of the form

$$Z = \sum_{i=1}^{K} \alpha_k \otimes \beta_k \otimes \gamma_k$$

where $\alpha_k \in \mathbb{R}^m$, $\beta_k \in \mathbb{R}^d$, $\gamma_k \in \mathbb{R}^n$ and the user factors, embedding factors and image factors respectively. The parameter K specifies the number of factors.

Letting $\alpha \in \mathbb{R}^{m \times K}$, $\beta \in \mathbb{R}^{d \times K}$ and $\gamma \in \mathbb{R}^{n \times K}$ denote the factors in matrix form, the image embedding matrix can be written as:

$$\begin{aligned} P^u &= \sum_{i=1}^{m} u_i Z^{(i)} \\ &= \sum_{i=1}^{m} u_i \left(\sum_{i=1}^{K} [\alpha_k]_i \beta_k \otimes \gamma_k \right) \\ &= \sum_{k=1}^{K} \left(\alpha_k^\top u \right) \beta_k \otimes \gamma_k \\ &= \beta \text{diag} \left(\alpha^\top u \right) \gamma^\top \end{aligned}$$

The user-multiplicative image embedding function can be re-written as:

$$\phi_{I \times U}(x, u) = P^u x = \beta \text{diag} \left(\alpha^\top u \right) \gamma^\top x$$

Recall from section 3.4 that in every iteration of training a gradient step is taken to minimize

$$L = |m - f(x, u, y^+) + f(x, u, y^-)|$$

where $f(x, u, y) = \phi_{I \times U}(x, u)^\top \phi_H(y)$ is a scoring function. Expanding the loss we get:

$$\begin{aligned} L &= |m - f(x, u, y^+) + f(x, u, y^-)| \\ &= |m - \phi_{I \times U}(x, u)^\top \phi_H(y^+) + \phi_{I \times U}(x, u)^\top \phi_H(y^-)| \\ &= |m - \beta \text{diag} \left(\alpha^\top u \right) \gamma^\top x (V_{y^+} - V_{y^-})| \end{aligned}$$

The gradient of L with respect to the user factors, embedding factors and image factors is:

$$\begin{aligned} \frac{\partial L}{\partial \alpha_k} &= - \left(\sum_{i=1}^{m} u_i \right) \beta \gamma^\top x (V_{y^+} - V_{y^-}) \\ \frac{\partial L}{\partial \beta} &= -\text{diag}(\alpha^\top u) \gamma^\top x (V_{y^+} - V_{y^-}) \\ \frac{\partial L}{\partial \gamma} &= - \left(\beta \text{diag}(\alpha^\top u) \right) \left(x (V_{y^+} - V_{y^-}) \right) \end{aligned}$$

B. DICTIONARY OF COMMON HASHTAGS

Many of the hashtags referenced in this work have non obvious meanings. Table 7 provides definitions of several of the more frequent hashtags.

Hashtag	Meaning
`#tbt`	throw back thursday
`#tb`	throw back
`#mcm`	man crush monday
`#mce`	man crush everyday
`#wcw`	woman crush wednesday
`#wce`	woman crush everyday
`#ss`	selfie sunday
`#rp`	repost
`#lmp`	like my post

Table 7: Definition of common hashtags.