Word2VisualVec: Cross-Media Retrieval by Visual Feature Prediction

Jianfeng Dong[†][*] Xirong Li[‡] Cees G. M. Snoek[§]
[†]Zhejiang University
[‡]Renmin University of China
[§]University of Amsterdam & Qualcomm Research Netherlands

ABSTRACT

This paper attacks the challenging problem of cross-media retrieval. That is, given an image find the text best describing its content, or the other way around. Different from existing works, which either rely on a joint space, or a text space, we propose to perform cross-media retrieval in a visual space only. We contribute *Word2VisualVec*, a deep neural network architecture that learns to predict a deep visual encoding of textual input. We discuss its architecture for prediction of CaffeNet and GoogleNet features, as well as its loss functions for learning from text/image pairs in large-scale click-through logs and image sentences. Experiments on the Clickture-Lite and Flickr8K corpora demonstrate the robustness for both Text-to-Image and Image-to-Text retrieval, outperforming the state-of-the-art on both accounts. Interestingly, an embedding in predicted visual feature space is also highly effective when searching in text only.

Categories and Subject Descriptors

H.3.3 [**INFORMATION STORAGE AND RETRIEVAL**]: Information Search and Retrieval

Keywords

Cross-media retrieval, Text embedding, Visual encoding

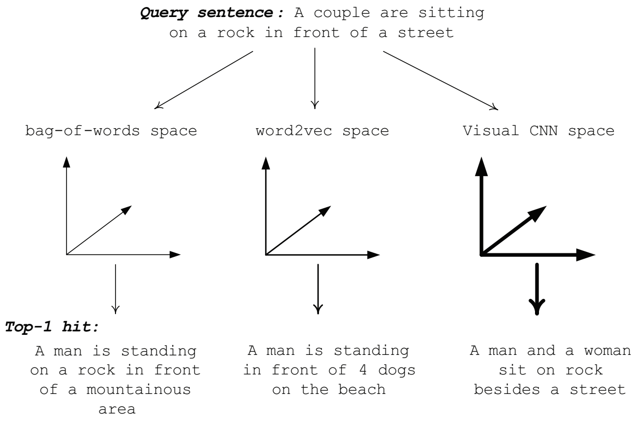

Query sentence: A couple are sitting on a rock in front of a street
bag-of-words space
word2vec space
Visual CNN space
Top-1 hit:
A man is standing on a rock in front of a mountainous area
A man is standing in front of 4 dogs on the beach
A man and a woman sit on rock besides a street

Figure 1: Our motivation. By embedding text into a visual feature space, we aim for a text representation that captures both semantic and visual similarities, compared to the classical bag-of-words and learned subspaces such as word2vec. The top-1 hit sentences are retrieved from 4,000 test sentences of the popular Flickr8K dataset [13].

1. INTRODUCTION

This paper attacks the challenging problem of cross-media retrieval. That is, given an image find the text best describing its content, or the other way around. Since image and text are of two distinct modalities, the common approach is to represent both of them in a common subspace, wherein the cross-media relevance between image and query is computed [2,8–10,19,33]. Recently, a distributional text embedding provided by Word2Vec [23] was found to be an effective space to perform cross-media retrieval [4,9,21,24]. Different from these existing works, which either rely on a joint space, or a text space, we propose to perform cross-media retrieval in a visual space only.

A text embedding in a visual space is potentially advantageous to the classical bag-of-words vector model and Word2Vec, as exemplified in Fig. 1. Consider searching for sentences describing a similar scene as '*A couple are sitting on a rock in front of a street*'. Neither of the bag-of-words space and the Word2Vec space returns a satisfying result, as the former is over rigid while the latter could be too compact to be discriminative. In contrast, a proper embedding in a good visual space can capture both semantic and visual similarities, finding a highly similar sentence as '*A man and a woman sit on rock besides a street*'.

We are inspired by recent success of deep convolutional neural networks in image categorization [16, 20, 30]. These neural networks learn a textual class prediction of an image by successive layers that perform convolutions, nonlinearities, pooling, fully connections, and big amounts of labeled image data, e.g., ImageNet [28]. The breakthrough was by Krizhevsky *et al.* [20] who demonstrated for the first time the supremacy of deep learning representation over

[*] Work performed at Renmin University of China.

shallow representations. More recently it was shown that the deeper these networks become, the better their class prediction ability [31]. What is more, the features derived from the layers of these networks are shown to be superior representations for various computer vision [27, 35] and multimedia retrieval [17, 18, 29] challenges. We also rely on a layered neural network architecture, but rather than predicting a textual class label for an image, we strive to predict a deep visual feature from a text string, let it be real user queries or natural language descriptions, for the purpose of cross-media retrieval.

Learning a representation for cross-media retrieval is not straightforward, especially as the equivalent of ImageNet does not exist. For cross-media retrieval in a joint subspace, feature transformation approaches such as Canonical Correlation Analysis [5] and Polynomial Semantic Indexing [1] are common. Having a given query represented by a bag-of-words vector and a given image represented by a low-level visual feature vector, both algorithms seek projection matrices to embed the query and the image into the space of lower dimensionality, as illustrated in Fig. 2(a). When relying on a Word2Vec space for cross-media retrieval, large-scale text corporate such as Wikipedia and Google News can be exploited. This is the tactic used by DeViSE [9] and ConSE [24]. Alternatively, Bai *et al.* propose a deep CNN model capable of extracting a bag-of-words vector from the image content, and thus conduct cross-media retrieval in a textual space. A novel element in their work is that they construct many relevant pairs of image and text from large click-through data [14], where image-text pairs associated with large user-click count are selected. The proposed Word2VisualVec is capable of learning from click-through data as well, and the resultant text representation is more effective than joint subspace representation, Word2Vec , and bag-of-words that have been exploited thus far.

Aiming for a new text representation for cross-media retrieval, this paper makes the following contributions:

- We propose *Word2VisualVec*, a deep neural network architecture that learns to predict a deep visual encoding of textual input, and thus enables cross-media retrieval in a visual space. Two loss functions, i.e., Mean Squared Error and Marginal Ranking Loss, are exploited to learn from text/image pairs in large-scale click-through logs and image sentences.

- Cross-media retrieval with *Word2VisualVec* outperforms state-of-the-art on the challenging Clickture-Lite dataset [14] for Text-to-Image retrieval and on the Flickr8k dataset [13] for Image-to-Text retrieval.

- We show Word2VisualVec is also a superior representation for text-only retrieval, when compared to the classical bag-of-words vector model and Word2Vec.

Before detailing our approach, we first highlight in more detail related work.

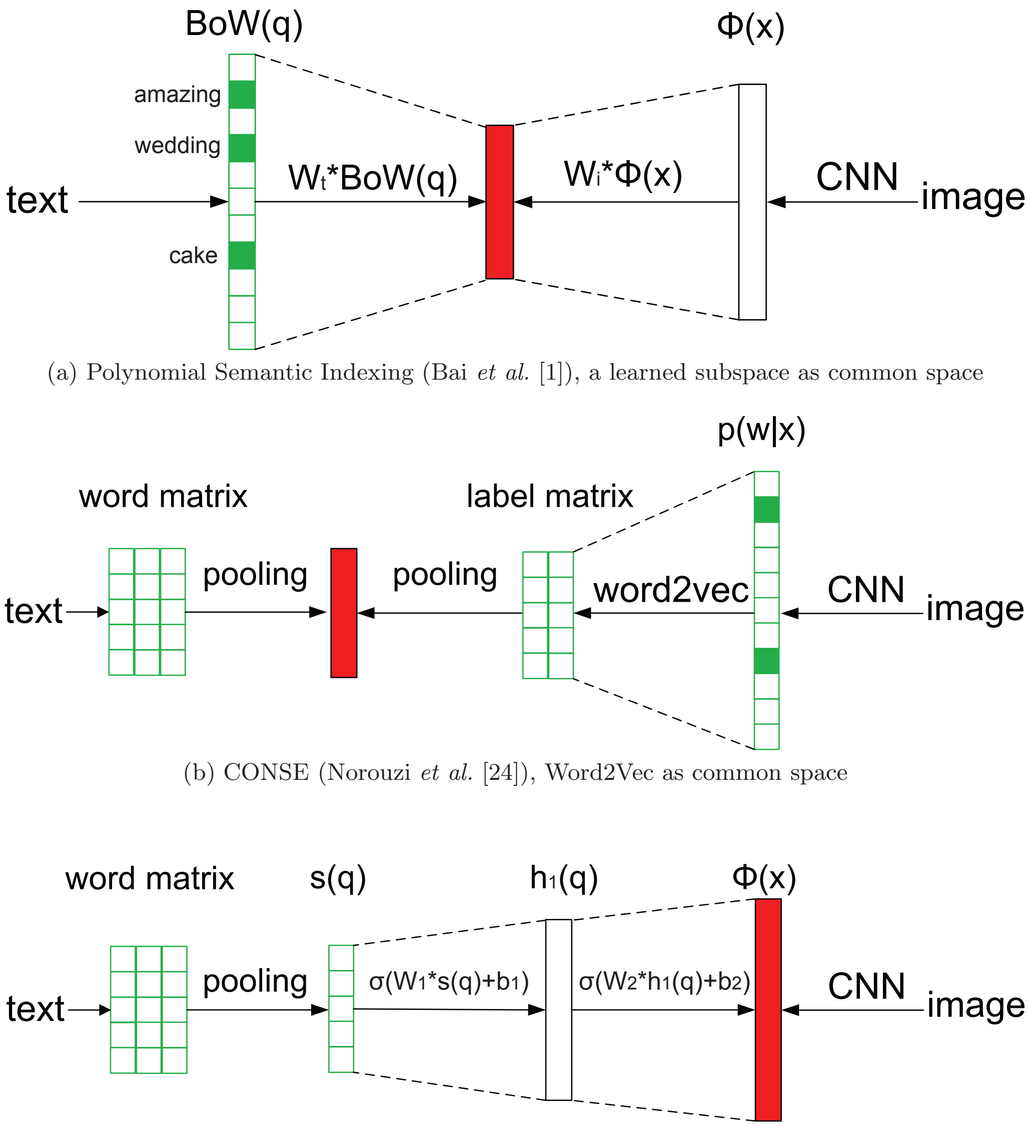
2. RELATED WORK

For embedding a text and an unlabeled image into a common space, what matters are forms of the embeddings and objectives to be optimized. So we review recent progresses in these two aspects.

Regarding the forms, the mainstream approach is to place affine transformations on both text and image sides to construct a latent subspace [1, 5, 11]. Depending on the choice of objectives, the embedding technique is known as Canonical Correlation Analysis (CCA) if one aims to maximize the correlation between embedding vectors of relevant pairs of query and image [5, 26], or as Polynomial Semantic Indexing (PSI) [1] if the objective is to minimize a marginal ranking loss. Gong *et al.* [10] extend CCA by proposing Stacked Auxiliary Embedding (SAE) to improve the joint text/image embedding, achieved by transferring knowledge from one million weakly tagged Flickr images. In [25] Pan *et al.* propose to minimize the distance of relevant pairs in the latent space, with regularization terms to preserve the inherent structure in each original space. A recent work by Yao *et al.* [34] considers a joint use of CCA and PSI, achieved by firstly finding a latent space by CCA and then re-adjusting the space to incorporate ranking preferences from click-through data. Habibian *et al.* [11] leverage a textual projection matrix and a visual projection matrix to embed both video and description into a latent subspace.

For the success of deep learning in computer vision and natural language processing, we observe an increase use of such techniques as an alternative to the affine transformation. In [36], for instance, Yu *et al.* use a deep Convolutional Neural Network (CNN) for image embedding, while at the same time keep the transformation at the text side. In the DeViSE model developed by Frome *et al.* [9], the common space is formed by a pre-trained Word2Vec model [23], where the embedding vector of a text is obtained by average pooling of the vectors of its words. In a follow-up work, Norouzi *et al.* employ Word2Vec for both text and image embedding [24]. In their CONSE model, an image is embedded into the Word2Vec space, achieved by a convex combination of the word embedding vectors of the visual labels predicted to be most relevant to the image. Bai *et al.* remove the text embedding part by using BoWDNN, a variant of AlexNet that outputs a bag-of-words vector for an input image [2]. The use of Word2Vec makes DeViSE and CONSE handle large vocabularies with ease. In contrast, BoWDNN has to predefine the size of the bag-of-words layer, so the amount of words the model can cover is relatively limited. Fang *et al.* [7] employ a word hashing technique [15] to vectorize a sentence before feeding it to a deep multimodal similarity model. In particular, a specific word is first decomposed into a list of letter-trigrams, e.g., *dog* as {#do, dog, og#}. Consequently, a sentence is represented by a letter-trigram counting vector. Since the number of unique letter-trigrams is less than the number of English words, it has a better scalability than bag-of-words. Nonetheless, its size remains relatively large when compared to Word2Vec. As shown in Fig. 2(c), the proposed Word2VisualVec is built on top of Word2Vec. Thus it inherits the merit of handling a large vocabulary. More importantly, different from the above works, Word2VisualVec embeds text into a visual space instead of an intermediate subspace.

In the context of image-to-text retrieval, there is an increasing interest in building a common space for cross-media similarity computation using Recurrent Neural Networks, see Karpathy and Fei-Fei [19] and Ma *et al.* [22]. The embedding is optimized by minimizing a marginal ranking loss [19] or a negative likelihood [22]. Again Word2VisualVec differs from these works in terms of the common space. Fur-



(a) Polynomial Semantic Indexing (Bai *et al.* [1]), a learned subspace as common space

(b) CONSE (Norouzi *et al.* [24]), Word2Vec as common space

(c) *Word2VisualVec* (this paper), a visual CNN feature space as common space

Figure 2: A conceptual illustration of varied *text embedding* models for cross-media similarity computation. The red shaded layer corresponds to the common space of the individual models. While existing works focus on learning an intermediate representation, the proposed *Word2VisualVec* model predicts visual CNN features from text (best viewed in color).

thermore, while RNN models count on high quality training data where text well describes image, Word2VisualVec has a capability of learning from weakly labeled training examples from click-through data, where text (namely real user queries) tends to be short and arbitrary.

3. OUR APPROACH

Our goal is to learn a visual representation of text. With this representation, cross-media relevance between a given text q and a specific unlabeled image x can be directly computed in a visual feature space.

More formally, let $\phi(x) \in \mathbb{R}^d$ be a d-dimensional visual feature vector. We aim for a text representation $r(q) \in \mathbb{R}^d$ such that the cross-media similarity can be expressed in terms of $\phi(x)$ and $r(q)$, say, in the form of an inner product.

Next, we present our choice of the visual feature space, followed by the proposed Word2VisualVec model.

3.1 Image Representation

Apart from its original mission of visual class recognition, a pretrained deep convolutional neural network (CNN) has now been recognized as an effective image feature extractor [27]. We follow this good practice, instantiating $\phi(x)$ with a CNN feature vector. In particular, we employ two pre-trained models, i.e., CaffeNet [16] and GoogLeNet [31]. Both of them were trained using millions of examples of

1,000 visual objects defined in the Large Scale Visual Recognition Challenge [28]. Although in theory each internal layer can be taken as a specific visual feature vector, we need $\phi(x)$ to capture both visual and semantic similarities. In that regard, we opt for the last fully connected layer (fc7) of CaffeNet and the pool5 layer of GoogLeNet, resulting in visual feature vectors of 4,096 and 1,024 dimensions, respectively.

3.2 The Word2VisualVec Model

As illustrated in Fig. 2(c), the proposed *Word2VisualVec* is a fully connected multi-layer neural net. This model essentially extracts visual CNN features, but from text.

Query text in real-world image retrieval tends to be short, the majority of which contain less than four words [14]. Extracting meaningful structures from such short text is difficult. Hence, we choose to first vectorize a text string by mean pooling over its words, i.e.,

$$s(q) := \frac{1}{|q|} \sum_{w \in q} w2v(w), \tag{1}$$

where $w2v(w)$ corresponds to individual word embedding vectors, and $|q|$ is the text length. Recent studies suggest that word2vec trained on Flickr tags better captures visual relationships than its counterpart learned from web documents [4, 21]. We therefore pre-train by the skip-gram algorithm a 500-dimensional word2vec model using English tags of 30 million Flickr images. The use of Word2Vec reduces the size of the input layer from hundreds of thousands to 500 only. This design choice substantially reduces the amount of parameters to be tuned.

The output of the first layer $s(q)$ goes through subsequent hidden layers until it reaches an output layer $r(q)$, which resides in the visual feature space. More concretely, by applying an affine transformation on $s(q)$, followed by an element-wise ReLU activation $\sigma(z) = \max(0, z)$, we obtain the first hidden layer $h_1(q)$ of an l-layer Word2VisualVec as

$$h_1(q) = \sigma(W_1 s(q) + b_1), \tag{2}$$

and the following hidden layers as

$$h_i(q) = \sigma(W_i h_{i-1}(q) + b_i), i = 2, ..., l-2, \tag{3}$$

where W_i parameterizes the affine transformation of the i-th hidden layer and b_i is a bias terms. In a similar manner we compute the output layer $r(q)$ as

$$r(q) = \sigma(W_l h_{l-1}(q) + b_l). \tag{4}$$

Putting together, the learnable parameters are represented by $\theta = [W_1, b_1, \dots, W_l, b_l]$.

Depending on the dimensionality of CNN features in use, we use distinct model architectures. In particular, for CaffeNet-fc7 we use a net of four layers, with the number of neurons per layer set to be 500, 1000, 2000, 4096 respectively. For GoogeLetNet-pool5, a three-layer net of 500-1000-1024 is deployed. We refer to Section 5.1 for a comprehensive evaluation of model design choices.

Cross-media relevance computation. Given a novel text, we obtain $r(q)$ by forward computation through the network. Subsequently the cross-media relevance between the text and a specific image x is computed as the cosine similarity between $r(q)$ and $\phi(x)$:

$$sim(q, x) = \frac{r(q) \cdot \phi(x)}{\|r(q)\| \, \|\phi(x)\|} \tag{5}$$

We choose this similarity as it normalizes feature vectors and is found to be better than the dot product.

3.3 Learning Algorithm for Word2VisualVec

In order to train the cross-media model, we need a training dataset $\mathcal{D} = \{(x, q)\}$, which consists of many relevant image-text pairs. Concerning the objective function, we consider two loss functions. One is Mean Squared Error (MSE), aiming to reconstruct $\phi(x)$ from q. Minimizing this loss often requires high-quality training instances, namely the text shall provide an accurate and rich description of the image's visual content. To alleviate the demand of such training data, the second loss function in consideration is Marginal Ranking Loss (MRL), which is in more prevalent use in the literature, see Table 1.

The MSE loss l_{mse} for a given training pair is defined as

$$l_{mse}(x, q; \theta) = (r(q) - \phi(x))^2. \tag{6}$$

We train the network to minimize the overall MSE loss:

$$\theta_{mse} = \underset{\theta}{\operatorname{argmin}} \sum_{(x,q) \in \mathcal{D}} w_{x,q} \cdot l_{mse}(x, q; \theta), \tag{7}$$

where $w_{x,q}$ is an optional weight reflecting the importance of a specific training pair. For instance, when learning from user-clicked data, an image-query pair getting more clicks are more likely to be mutually relevant, and thus more important in the training process.

As for MRL, its ranking based objective function requires triplets as input. A triplet extension of $\mathcal{D}$ is denoted as $\mathcal{D}_{tri}$. Formats of the triplets vary over tasks. For text-to-image retrieval, we need to optimize image ranking for a given text, so a triplet has a form of $(q, x^+, x^-) \in \mathcal{D}_{tri}$, where x^+ and x^- indicate images relevant and irrelevant with respect to q. For image-to-text retrieval, a triplet is defined as (x, q^+, q^-), where image x is relevant to q^+ but irrelevant to q^-. In the following, we describe only the training process for text-to-image, as image-to-text can be trained in a dual form.

The MRL loss for a specific training triplet is defined as

$$l_{mrl} = \max\{0, 1 + \cos(r(q), \phi(x^-)) - \cos(r(q), \phi(x^+))\} \tag{8}$$

where $\cos(\cdot, \cdot)$ is the cosine similarity between two vectors. To train Word2VisualVec with the MRL loss, we optimize

$$\theta_{mrl} = \underset{\theta}{\operatorname{argmin}} \sum_{(q,x^+,x^-) \in \mathcal{D}_{tri}} l_{mrl}(q, x^+, x^-; \theta) \tag{9}$$

Optimization. We solve both Eq. (7) and Eq. (9) using stochastic gradient descent with RMSprop [32]. This optimization algorithm rescales the learning rate for each parameter according to the history of the gradients for that parameter. In particular, the algorithm divides the learning rate by an exponentially decaying average of squared gradients, which prevents the learning rate from effectively shrinking over time. We empirically set the initial learning rate $\eta = 0.001$, decay weights $\gamma = 0.9$ and small constant $\epsilon = 10^{-06}$ for RMSprop. In addition, we apply dropout to the hidden layer, as this trick is empirically found to be effective to prevent co-adaption of feature detectors [20] and thus helpful for mitigating model over-fitting. Lastly, we apply an early stop strategy: stop training if there is no performance improvement on the validation set in successive 10 epochs, with the maximal number of epochs set to be 200.

Table 1: A quick summary of cross-media retrieval models compared in this work.

Model	Common space	Dimensionality	Loss function	No. parameters	Tasks
CONSE [24]	word2vec	500	–	1	text-to-image
DeViSE [9]	word2vec	500	marginal ranking loss	2 millions	text-to-image, image-to-text
BoWDNN [2]	bag-of-words	50,000	cosine distance	183 millions	text-to-image
PSI [26]	learned subspace	1,000	marginal ranking loss	54 millions	text-to-image
CCA [12]	learned subspace	–	reduced rank regression	–	text-to-image
RCCA [34]	learned subspace	80	marginal ranking loss	4 millions	text-to-image
SAE [10]	learned subspace	96	reduced rank regression	2 millions	image-to-text
BRNN [19]	learned subspace	1,000	marginal ranking loss	5 millions	image-to-text
m-RNN [22]	learned subspace	512	negative log-likelihood	2 millions	image-to-text
Word2VisualVec-mse	image feature space	1024/4096	mean squared error	1.5 million / 10 millions	text-to-image, image-to-text
Word2VisualVec-mrl	image feature space	1024/4096	marginal ranking loss	1.5 million / 10 millions	text-to-image, image-to-text

4. EXPERIMENTAL SETUP

In order to verify the viability of Word2VisualVec for cross-media retrieval, we perform experiments in the following three tasks, i.e., Text-to-Image, Image-to-Text, and Text-to-Text. The Text-to-Image task is to retrieve images relevant with respect to a given textual query from an (unlabeled) image collection. The second task is to retrieve sentences that best describe a given image from a sentence collection. Finally, in Text-to-Text we investigate whether Word2VisualVec can find texts that are semantically and visually close to a given text. We setup evaluations with respect to each task as follows. Cross-media retrieval models evaluated in our experiments are summarized in Table 1.

4.1 Task 1. Text-to-Image Retrieval

Data. We adopt Clickture-Lite [14], a benchmark dataset for image retrieval with real user queries. The dataset consists of two parts: a ground-truthed development set of 80K images and a training set of 1M images.

The development set contains 1K real user queries, 79,665 images, and 79,926 query-image pairs for cross-media similarity computation. Each pair is labeled as Excellent, Good or Bad, according to the image's relevance with respect to the query. Similar to previous works [2,6,26,34], we use this set as the test set, enabling a head-to-head comparison to the state-of-the-art.

The training set consists of 23M click-through triads of (query, image, *click*), where *click* is the accumulated amount of user clicks a specific image has received with respect to a given query. An image corresponds with to one or more triads, and a query may also appear in multiple triads that are associated with different images. We randomly split the training images into two disjoint subsets, 90% for training and 10% for validation. Simple query preprocessing has been applied, i.e., removing punctuation and lemmatizing words by the NLTK toolkit [3]. Meaningless words such as "image" and "picture" and standard English stopwords are also removed.

Evaluation criterion. Following the protocol [14], we compute Normalized Discounted Cumulated Gain (NDCG) at the rank of 25. The overall performance is measured by averaging NDCG scores over the test queries.

4.2 Task 2. Image-to-Text Retrieval

The second experiment is conducted in the image-to-text retrieval task. Given a test image, a provided list of sentences is sorted in descending order in terms of image-sentence relevance scores.

Data. We use Flickr8k [13], a popular benchmark set for image-to-text retrieval. The set contains 8K images collected from Flickr. Each image is associated with five crowd-sourced English sentences, which briefly describe main objects and scenes present in the image. We follow a standard data partition [19]: 6K images for training, 1K images for validation, and the remaining 1K images for test.

Evaluation criteria. We report rank-based performance metrics, namely $R@K$ ($K = 1, 5, 10$) and Median rank ($Med\,r$) as previous works [10,19,22]. In particular, $R@K$ computes the percentage of test images for which at least one correct result is found among the top-K retrieved sentences, and $Med\ r$ is the median rank of the first correct result in the ranking list. Hence, higher $R@K$ and lower $Med\ r$ means better performance.

4.3 Task 3. Text-to-Text Retrieval

This experiment conducts text-only retrieval, where candidate texts are sorted in descending order according to their relevance scores to a given text. We predict deep visual feature vectors for both the input query text and the text in the test set. Relevance scores between text are computed as the cosine similarity between the corresponding vectors.

Data. We need pairs of texts that are visually and semantically relevant. Image captions in Flickr8K meet this requirement as they are meant for describing the same visual content, and thus visually relevant. Moreover, since they were independently written by distinct users, the wording may vary across the users, requiring a text representation to capture shared semantics among distinct words. Consider for instance the following two sentences captioning the same image: "*Boys in life jackets on a watercraft*" and "*Several young people in life jackets are sitting on something floating in water*". In that regard, we construct a test set for text-to-text retrieval using the Flickr8K test set. In particular, for each image we treat its first sentence as a query and the other four sentences as test instances, resulting 1K queries and a test pool of 4K sentences.

Table 2: Performance of Word2VisualVec with text projected to different CaffeNet layers. Task: Text-to-Image. Loss function: MSE.

CaffeNet layer	**Word2VisualVec net architecture**	NDCG_{25}
pool5	500-1000-2000-9216	0.4965
fc6	500-1000-2000-4096	0.4954
fc7	500-1000-2000-4096	**0.5107**

Table 3: Performance of Word2VisualVec as the number of net layers increases. Task: Text-to-Image. Loss function: MSE.

Net architecture	NDCG_{25}
2 layers: 500-4096	0.5008
3 layers: 500-1000-4096	0.5075
4 layers: 500-1000-2000-4096	0.5107
5 layers: 500-1000-2000-3000-4096	**0.5119**

Evaluation criterion. We report mean Average Precision (mAP).

5. EXPERIMENTS

5.1 Experiment 1. Properties of Word2VisualVec

In this experiment, we investigate the impact of major design choices, e.g., visual CNN features, net architecture and loss functions, on the performance of Word2VisualVec. Due to high complexity of the problem, evaluating all the variables simultaneously will be computationally prohibitive. Therefore, the evaluation is conducted sequentially, focusing on one variable per time, with others fixed.

What deep CNN? First of all, concerning the choice of CNN models, we find that CaffeNet works better in the text-to-image scenario, but is outperformed by GoogeLeNet in the image-to-text scenario. So in what follows, we use CaffeNet for text-to-image retrieval and GoogleNet for image-to-text and text-to-text retrieval.

Which visual space? Table 2 shows the performance of Word2VisualVec with text projected to the pool5, fc6, and fc7 layers of CaffeNet, respectively. The result confirms our hypothesis in Section 3 that using the fc7 layer as visual CNN feature vectors is a good choice.

Model architecture of Word2VisualVec. Table 3 shows the performance of Word2VisualVec with respect to changes in the number of net layers. As more layers are in use, Word2VisualVec improves, at the cost of increasing learning complexity. As a trade-off between accuracy and efficiency, we use a net architecture of 500-1000-2000-4096 for text-to-image retrieval. In a similar manner we identify a net of 500-1000-1024 for the other two tasks.

Word2Vec vs Hashing. Concerning the choice of word embedding for the first layer, we compare Word2Vec and word hashing. Compared to bag-of-words, word hashing has the advantage of reducing the size of the input layer while generalizing well to infrequent and unseen words [15]. Following [7], we convert each word in a given text to a letter-trigram count vector, resulting in an input layer of 4,402 nodes. As shown in Table 4, Word2Vec is on par with word hasing, yet maintains the advantage of a more compact input layer.

Table 4: The influence of word embedding for the input layer. Task: Image-to-Text. Loss function: MSE.

First layer	R@1	R@5	R@10	Med r
Word hashing	25.7	51.0	**63.2**	5
Word2Vec	25.7	**51.6**	63.1	5

Table 5: Performance of *Word2VisualVec* with distinct loss functions.

	Text-to-Image	**Image-to-Text**			
Loss	NDCG@25	R@1	R@5	R@10	Med r
MSE	0.5107	**25.7**	**51.6**	**63.1**	**5**
MSE + ClickCount	0.5128	–	–	–	–
MRL	0.5129	21.0	43.1	55.9	8
MRL + $\text{Initialize}_{\text{MSE}}$	**0.5145**	23.4	50.4	62.1	5

What loss function? Finally, we evaluate the influence of using different loss functions. The performance is shown in Table 5. The choice of loss functions turns out to be task dependent. For text-to-image retrieval, since each training pair of query and image is associated with a user click count, we tried adding the (log-scaled) click count as weights when minimizing MSE. The better performance of MSE + ClickCount against MSE shows the benefit of the click count information. The best performance is achieved by MRL + $\text{Initialize}_{\text{MSE}}$, which first minimizes MSE to obtain a better parameter initialization as an alternative to random initialization, and then minimizes MRL. Interestingly, the top performer for image-to-text is Word2VisualVec with the MSE loss. This is probably due to distinct characteristics of texts in the two tasks. Recall that texts in text-to-image are real queries sampled from a web image search query log. Over 70% of the queries contain less than five words [14], and often lack correspondence to specific visual appearance. This property makes the reconstruction of CNN vectors more challenging, when compared to reconstructing from sentences in the second task, which are meant for describing the visual content.

Next, we compare Word2VisualVec with state-of-the-art in the individual tasks. For a fair comparison, we take numbers directly from the original papers whenever applicable.

5.2 Experiment 2. Text-to-Image Retrieval

We compare six state-of-the-art works that implement image and text embedding in varied manners. They are:
1) CONSE [24]. Originally proposed for zero-shot image labeling, applied to text-to-image retrieval by [6].
2) CCA [26]. Projecting text and image into a learned subspace by two affine transformations, with the goal of maximizing the correlation between embedding vectors of relevant image-text pairs.
3) PSI [1]. Originally introduced for document retrieval,

Table 6: Performance of text-to-image retrieval. Existing works marked with stars (*) are our implementation.

Model	NDCG$_{25}$
Random ranker	0.4702
CONSE [6]	0.4857
PSI* [1]	0.5016
DeViSE* [9]	0.5033
CCA [26]	0.5055
BoWDNN [2]	0.5089
RCCA [34]	0.5112
Word2VisualVec	**0.5145**

applied to text-to-image by [26].
4) DeViSE [9]. Building a common subspace on top of CNN features of image and Word2Vec features of query.
5) BoWDNN [2]. This CNN model takes as input an unlabeled image and outputs a bag-of-words vector.
6) RCCA [34]. An improved version of CCA, first learning a common space by CCA and then adjusting the space to preserve preference relationships in click-through data.

In addition to the chosen models, we also report a random baseline (random ranker), which sorts images at random. The performance of the seven models is presented in Table 6. All the models outperform the random baseline. It is worth pointing out that finding images from an unlabeled and unconstrained image collection for thousands of real-world user queries is a grand challenge. Hence, although the performance divergence may appear to be relatively small, the significance of the individual models shall not be underestimated.

Comparing the individual models, by learning from the click-through data, PSI and DeViSE outperform the fully unsupervised CONSE. Recall that the main difference between PSI and DeViSE is that the latter uses Word2Vec as an alternative to the bag-of-words vector. So the better performance of DeViSE suggests that Word2Vec provides a better starting point for vectorizing queries. RCCA is better than CCA, as it additionally preserves preference relationships in click-through data on the basic of CCA. With NDCG$_{25}$ of 0.5145, Word2VisualVec outperforms all the competitors.

For a more intuitive understanding, we find neighbor queries of a given query with cosine similarity computed using bag-of-words, Word2Vec, and Word2VisualVec, respectively, as shown in Table 7. Consider query *giant sea monster*, bag-of-words and Word2Vec return weakly relevant queries such as *monster* and *giant*. By contrast, Word2VisualVec returns queries that are visually closer, such as *biggest shark in world* and *world biggest shark*. The result suggests that Word2VisualVec can find text queries depicting similar visual patterns, even query words differ substantially.

Table 7: Nearest queries retrieved in terms of the cosine distance between text vectors generated by bag-of-words, Word2Vec and *Word2VisualVec*, respectively

Query	**Neighboring Queries**		
	bag-of-words	*Word2Vec*	*Word2VisualVec*
giant sea monster	giant	monster sign	biggest shark in world
	sea	monster	world biggest shark
	monster	giant	shark attack
	ny giant	skylanders giant	real mermaid
	giant panda	scary monster	mermaid body found
fish taco	taco	taco	taco
	fish	fish	mexico food
	cartoon fish	big fish	dinner
	tropical fish	colorful fish	mexcian food
	bass fish	sea fish	bacon
cell phone	cell	cell	cellphone
	phone	phone	phone
	i phone	cell wall	i phone 5
	phone wallpaper	cellphone	iphone 5
	animal cell	eukaryotic cell	ipod touch

Table 8 shows image retrieval results produced by our model. As we can see, the visual content of the test images is quite diverse. Also because the data was crawled from the web, it contains some near-duplicate images, see the response of query 'fat cat'. Word2VisualVec is effective for handling queries with clear visual cues. Consider query '*woman bicycle*' for instance. Our model returns images of woman-style bicycle and images containing both woman and bicycle. Although no correct results are found for query '*how to merge two images in photoshop*', the model tends to find pictures of two persons. These qualitative results demonstrate the effectiveness of Word2VisualVec for text-to-image retrieval.

5.3 Experiment 3. Image-to-Text Retrieval

Besides DeViSE, we compare three more recent works on image-to-text retrieval, namely
1) SAE [10]. An improved version of kernel CCA, exploiting extra knowledge derived from one million weakly tagged Flickr image.
2) m-RNN [22]. Employing a RNN to embed both sentences and images into a common subspace, and ranking the sentences by a normalized posterior probability.
3) BRNN [19]. Employing a bi-directional RNN for a joint embedding of image and text, and ranking the sentences by relevance scores computed in terms of embedding vectors.

Performance of the five models is provided in Table 9. Word2VisualVec scores the best under multiple evaluation criteria. Recurrent Neural Networks based approaches, as exemplified by m-RNN and BRNN, are dominating the literature for image-to-text retrieval. This result shows that with proper text embedding, our non-RNN approach is also competitive. Also note that the predicted visual features of Word2VisualVec can easily be embedded in an RNN framework as well.

Some image-to-text results by Word2VisualVec is given in Table 10.

5.4 Experiment 4. Text-to-Text Retrieval

While Table 7 has provides some qualitative results demonstrating the advantage of Word2VisualVec against bag-of-

Table 8: Text-to-Image results produced by the proposed *Word2VisualVec* . Hand-picked queries are given in the first column. For a given test query, we sort the 80K test images in terms of their cross-media relevance scores with respect to the query, and consequently show the top-5 most similar images (second column) and the top-5 most dissimilar images (third column).

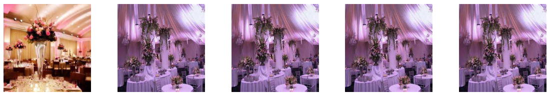
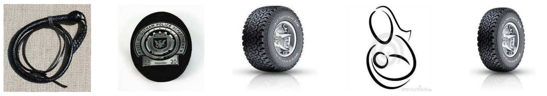
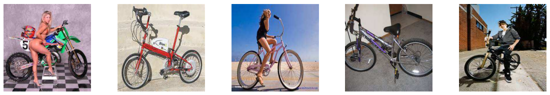
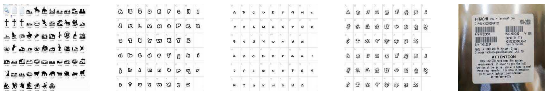
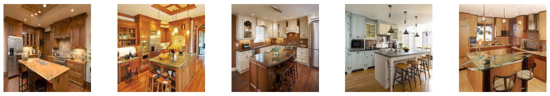
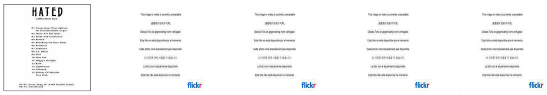
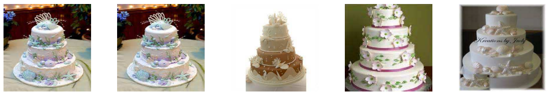
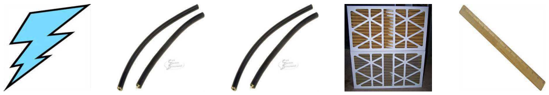
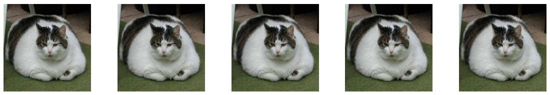
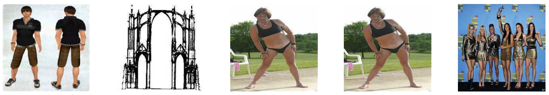
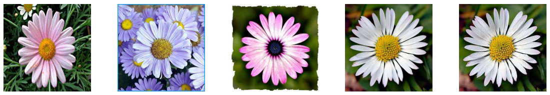
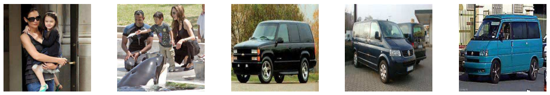
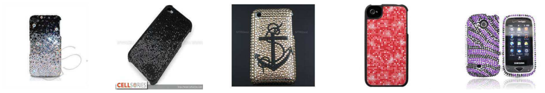
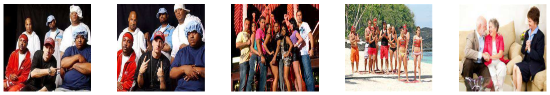
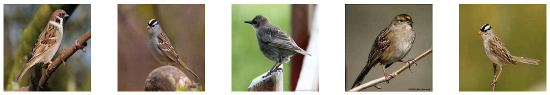
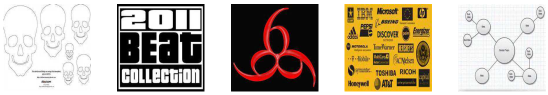
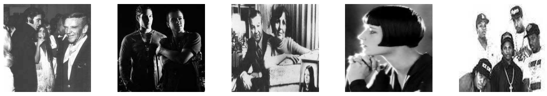
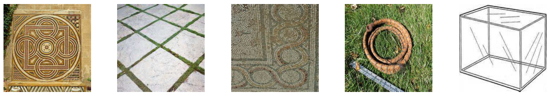
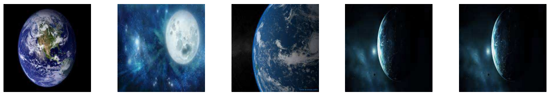
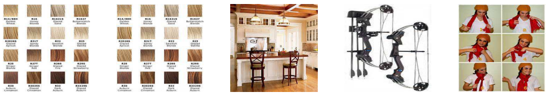
Test query	Top-5 most similar images	Top-5 most dissimilar images
wedding reception idea		
woman bicycle		
large kitchen design with island		
amazing wedding cake		
fat cat		
daisy flower		
bling phone case		
sparrow		
how to merge two image in photoshop		
earth from space wallpaper		

Table 9: Performance of image-to-text retrieval. Existing works marked with stars (*) are our implementation.

Model	R@1	R@5	R@10	Med r
DeViSE* [9]	8.0	22.7	33.1	21
SAE [10]	–	–	38.2	–
m-RNN [22]	14.5	37.2	48.5	11
BRNN [19]	16.5	40.6	54.2	7.6
Word2VisualVec	**25.7**	**51.6**	**63.1**	**5**

words and Word2Vec, this experiment further gives a quantitative evaluation.

Every sentence is vectorized respectively by bag-of-words, mean pooling of Word2Vec, and Word2VisualVec which represents the sentence in the visual CNN feature space. The cosine distance is applied to generate sentence rankings. Table 11 shows performance of the three text representations. With a clear margin, Word2VisualVec surpasses bag-of-words and Word2Vec. Since relevant pairs of sentences describe the same visual content, but with words vary over the sentences (see Table 10), the result allows us to conclude that Word2VisualVec better captures both visual and semantic similarities.

6. CONCLUSIONS

We present in this paper *Word2VisualVec*, a new representation for cross-media retrieval. Given a text string, let it be real user queries or natural language descriptions, Word2VisualVec represents the text by a deep visual feature. As a consequence, cross-media retrieval is performed

Table 10: Image-to-Text results produced by the proposed *Word2VisualVec* . Sentences outside the ground truth of each test image are marked with ✘.

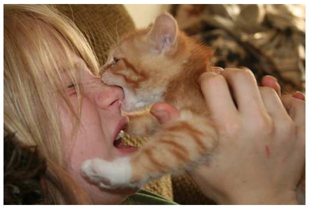
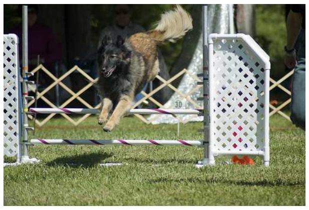
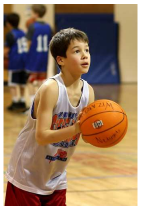
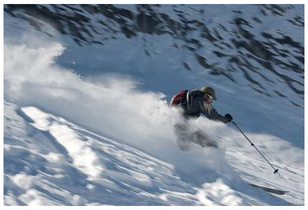
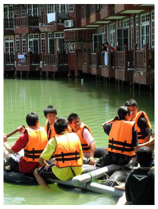
Test image	Top-5 retrieved sentences
	✔ Orange striped kitten biting blonde girl on the nose ✔ A orange kitten biting the nose of a child ✔ A blonde child is being bitten on the nose by a little orange kitten ✔ A cat bites a humans nose ✘ Two brown bears touching, face to face, with mouths open
	✔ A dog jumps over an obstacle ✘ A dog is jumping across an obstacle ✔ A dog jumps over a hurdle in the grass ✘ A dog jumps over a read and blue hurdle ✘ A dog jumps over a hurdle at a competition
	✘ A basketball game ✘ The Miami basketball player is looking ✘ The basketball player in white holds the ball ✘ The boy is playing basketball ✘ Miami basketball player shooting
	✘ A skier about to go down the mountain ✔ A man is skiing down a mountain ✘ A mountain skier heads down a mountain ✘ a skier dressed in black speeds down the mountain ✘ A skier catches air over the snow
	✔ Boys in life jackets on a watercraft ✔ A group of people on a boat ✔ A group of 6 boys are wearing yellow life vests and are on a make-shift raft ✔ Several young people in life jackets are sitting on something floating in water ✘ Several people are rowing a boat while being cheered on by a person with a drum

Table 11: Performance of text-to-text retrieval.

Text representation	mAP
bag-of-words	16.2
Word2Vec	22.5
Word2VisualVec	**34.6**

in the deep visual space only. Experiments on the challenging Clickture-Lite and Flickr8k datasets support our major conclusions as follows.

First, the *Word2VisualVec* representation better captures both semantic and visual similarities when compared to the classical bag-of-words vector model and distributional text embedding provided by Word2Vec. Second, with a proper loss function *Word2VisualVec* can be trained either using noise training instances automatically acquired from large-scale click-through data or using high-quality training data wherein image is well described by associated text. More specifically, Marginal Ranking Loss shall be used in the former case, while Mean Squared Error is the choice of the latter case. *Word2VisualVec* outperforms the state-of-the-art for Text-to-Image and Image-to-Text retrieval.

7. REFERENCES

[1] B. Bai, J. Weston, D. Grangier, R. Collobert, K. Sadamasa, Y. Qi, C. Cortes, M. Mohri, B. Bai, and J. Weston. Polynomial semantic indexing. In *Proc. of NIPS*, 2009.

[2] Y. Bai, W. Yu, T. Xiao, C. Xu, K. Yang, W.-Y. Ma, and T. Zhao. Bag-of-words based deep neural network for image retrieval. In *Proc. of ACM MM*, 2014.

[3] S. Bird, E. Loper, and E. Klein. *Natural Language Processing with Python*. O'Reilly Media Inc, 2009.

[4] S. Cappallo, T. Mensink, and C. G. M. Snoek. Image2emoji: Zero-shot emoji prediction for visual media. In *Proc. of ACM MM*, 2015.

[5] J. Costa Pereira, E. Coviello, G. Doyle, N. Rasiwasia, G. R. Lanckriet, R. Levy, and N. Vasconcelos. On the role of correlation and abstraction in cross-modal multimedia retrieval. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36(3):521–535, 2014.

[6] J. Dong, X. Li, S. Liao, J. Xu, D. Xu, and X. Du. Image retrieval by cross-media relevance fusion. In *Proc. of ACM MM*, 2015.

[7] H. Fang, S. Gupta, F. Iandola, R. K. Srivastava, L. Deng, P. Dollár, J. Gao, X. He, M. Mitchell, J. C. Platt, et al. From captions to visual concepts and back. In *Proc. of CVPR*, 2015.

[8] F. Feng, X. Wang, and R. Li. Cross-modal retrieval with correspondence autoencoder. In *Proc. of ACM MM*, 2014.

[9] A. Frome, G. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov. DeViSE: A deep visual-semantic embedding model. In *Proc. of NIPS*, 2013.

[10] Y. Gong, L. Wang, M. Hodosh, J. Hockenmaier, and S. Lazebnik. Improving image-sentence embeddings using large weakly annotated photo collections. In *Proc. of ECCV*, 2014.

[11] A. Habibian, T. Mensink, and C. G. Snoek. VideoStory: A new multimedia embedding for few-example recognition and translation of events. In *Proc. of ACM MM*, 2014.

[12] D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor. Canonical correlation analysis: An overview with application to learning methods. *Neural computation*, 16(12):2639–2664, 2004.

[13] M. Hodosh, P. Young, and J. Hockenmaier. Framing image description as a ranking task: Data, models and evaluation metrics. *J. Artif. Int. Res.*, 47(1):853–899, 2013.

[14] X.-S. Hua, L. Yang, J. Wang, J. Wang, M. Ye, K. Wang, Y. Rui, and J. Li. Clickage: Towards bridging semantic and intent gaps via mining click logs of search engines. In *Proc. of ACM MM*, 2013.

[15] P.-S. Huang, X. He, J. Gao, L. Deng, A. Acero, and L. Heck. Learning deep structured semantic models for web search using clickthrough data. In *Proc. of CIKM*, 2013.

[16] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *CoRR*, arXiv:1408.5093, 2014.

[17] L. Jiang, S.-I. Yu, D. Meng, Y. Yang, T. Mitamura, and A. Hauptmann. Fast and accurate content-based semantic search in 100m Internet videos. In *Proc. of ACM MM*, 2015.

[18] X. Jiang, F. Wu, X. Li, Z. Zhao, W. Lu, S. Tang, and Y. Zhuang. Deep compositional cross-modal learning to rank via local-global alignment. In *Proc. of ACM MM*, 2015.

[19] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proc. of CVPR*, 2015.

[20] A. Krizhevsky, I. Sutskever, and G. Hinton. ImageNet classification using deep convolutional neural networks. In *Proc. of NIPS*, 2012.

[21] X. Li, S. Liao, W. Lan, X. Du, and G. Yang. Zero-shot image tagging by hierarchical semantic embedding. In *Proc. of SIGIR*, 2015.

[22] J. Mao, W. Xu, Y. Yang, J. Wang, Z. Huang, and A. Yuille. Deep captioning with multimodal recurrent neural networks (m-RNN). In *Proc. of ICLR*, 2015.

[23] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Proc. of NIPS*, 2013.

[24] M. Norouzi, T. Mikolov, S. Bengio, Y. Singer, J. Shlens, A. Frome, G. Corrado, and J. Dean. Zero-shot learning by convex combination of semantic embeddings. In *Proc. of ICLR*, 2014.

[25] Y. Pan, T. Yao, T. Mei, H. Li, C.-W. Ngo, and Y. Rui. Click-through-based cross-view learning for image search. In *Proc. of SIGIR*, 2014.

[26] Y. Pan, T. Yao, X. Tian, H. Li, and C.-W. Ngo. Click-through-based subspace learning for image search. In *Proc. of ACM MM*, 2014.

[27] A. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. CNN features off-the-shelf: An astounding baseline for recognition. In *Proc. of CVPR DeepVision Workshop*, 2014.

[28] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. Berg, and L. Fei-Fei. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vision*, 115(3):211–252, 2015.

[29] X. Shang, H. Zhang, and T.-S. Chua. Deep learning generic features for cross-media retrieval. In *Proc. of MMM*, 2016.

[30] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[31] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proc. of CVPR*, 2015.

[32] T. Tieleman and G. Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning, 2012.

[33] F. Wu, X. Lu, Z. Zhang, S. Yan, Y. Rui, and Y. Zhuang. Cross-media semantic representation via bi-directional learning to rank. In *Proc. of ACM MM*, 2013.

[34] T. Yao, T. Mei, and C.-W. Ngo. Learning query and image similarities with ranking canonical correlation analysis. In *Proc. of ICCV*, 2015.

[35] G. Ye, Y. Li, H. Xu, D. Liu, and S.-F. Chang. EventNet: A large scale structured concept library for complex event detection in video. In *Proc. of ACM MM*, 2015.

[36] W. Yu, K. Yang, Y. Bai, H. Yao, and Y. Rui. Learning cross space mapping via dnn using large scale click-through logs. *IEEE Trans. Multimedia*, 17(11):2000–2007, 2015.