Cross-Media Similarity Evaluation for Web Image Retrieval in the Wild

Jianfeng Dong, Xirong Li*, and Duanqing Xu

Abstract—**In order to retrieve unlabeled images by textual queries, cross-media similarity computation is a key ingredient. Although novel methods are continuously introduced, little has been done to evaluate these methods together with large-scale query log analysis. Consequently, *how far have these methods brought us in answering real-user queries* is unclear. Given baseline methods that compute cross-media similarity using relatively simple text/image matching, *how much progress have advanced models made* is also unclear. This paper takes a pragmatic approach to answering the two questions. Queries are automatically categorized according to the proposed query visualness measure, and later connected to the evaluation of multiple cross-media similarity models on three test sets. Such a connection reveals that the success of the state-of-the-art is mainly attributed to their good performance on visual-oriented queries, while these queries account for only a small part of real-user queries. To quantify the current progress, we propose a simple text2image method, representing a novel test query by a set of images selected from large-scale query log. Consequently, computing cross-media similarity between the test query and a given image boils down to comparing the visual similarity between the given image and the selected images. Image retrieval experiments on the challenging Clickture dataset show that the proposed text2image compares favorably to recent deep learning based alternatives.**

Index Terms—**Web image retrieval, real-user query, cross-media similarity computation.**

I. Introduction

SINCE the early 1990s how to retrieve *unlabeled* images by textual queries has been a grand challenge in multimedia retrieval, and remains hot to this day [1]–[4]. In order to understand and exploit the interplay between visual content, textual query and user behaviors, web image retrieval demands multi-modal approaches [5]–[7] and thus makes it right at the heart of the multimedia field. As image and query are two distinct modalities, a cross-media similarity metric that effectively reflects image-query relevance is essential.

The key to cross-media similarity computation is to represent both image and query in a common space [9]–[14]. Although the idea seems to be simple, constructing a proper common space is non-trivial. A desirable property is to let each of the many queries stay closer to images relevant w.r.t. the queries than irrelevant images. In [12] for instance, Yu *et al.* employ deep neural network to embed images and queries into a joint latent space using large scale click-through logs. While in [13], Deng *et al.* firstly use dictionary learning to learn discriminative dictionary for images and queries respectively, then map the sparse codes of images and queries into a common label space by linear transformation.

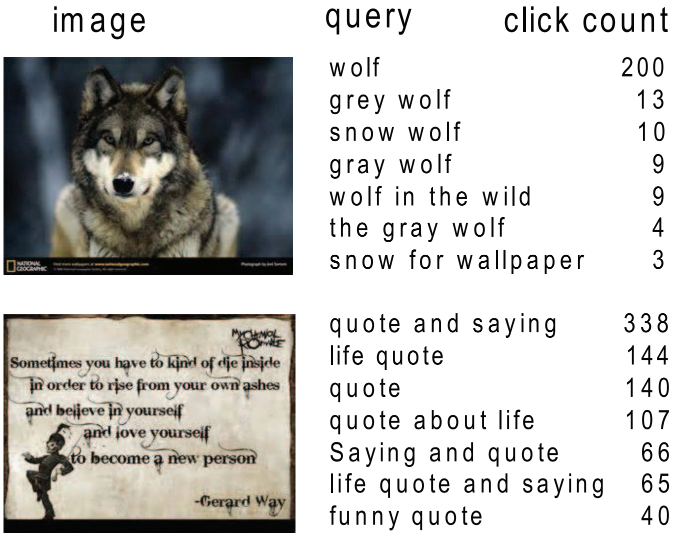


image	query	click count
	wolf	200
	grey wolf	13
	snow wolf	10
	gray wolf	9
	wolf in the wild	9
	the gray wolf	4
	snow for wallpaper	3
	quote and saying	338
	life quote	144
	quote	140
	quote about life	107
	Saying and quote	66
	life quote and saying	65
	funny quote	40

Sometimes you have to kind of die inside
in order to rise from your own ashes
and believe in yourself
and love yourself
to become a new person
-Gerard Way

Fig. 1. **Image, query and their user-click count**, sampled from query log data of a web image search engine [8].

Depending on the choice of the common space, we see three lines of research. One, represent images in a textual space by image auto-annotation [15]–[18] and consequently compute the cross-media similarity by text matching. Two, represent queries in a visual space by learning visual templates of the queries [19], [20], and compute the cross-media similarity by visual matching. Third, also the main stream, build a common latent space, e.g., by maximizing correlation between relevant image-query pairs [10], [11], [13], [21] or by minimizing a ranking based loss [7], [9], [12], [22].

Given the progress described above, an important question then is

> *Q1. how far have cross-media similarity models brought us in answering user queries in the wild?*

By wild we mean any textual query one might submit to a generic web image search engine. Such queries, due to their unconstrained nature, do not necessarily follow syntax, may contain typos, and more importantly, often lack correspondence to specific visual appearance, see Fig. 1. In spite of the rich literature, we argue that this question is largely untouched, presumably due to the lack of related benchmark data.

*Corresponding author. J. Dong and D. Xu are with the College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China. (e-mail: danieljf24@zju.edu.cn; xdq@zju.edu.cn) X. Li is with the Key Lab of Data Engineering and Knowledge Engineering, Renmin University of China, Beijing 100872, China. (e-mail: xirong@ruc.edu.cn)

For evaluating text-based image retrieval, we notice two types of benchmark data, i.e., tag-based [23], [24] and sentence-based [25], [26]. MIRFlickr [23] and NUS-WIDE [24] are two leading benchmark sets for tag-based image retrieval, providing 14 and 81 test tags respectively. These tags were chosen mainly because of their availability on Flickr. In contrast to such single-tag queries, in Flickr8k [25] and MSCOCO [26], a query is represented in the form of a sentence, which is meant for describing in brief main objects and scenes in an image. The wiki text-image dataset [27] is an extreme case of sentence-based datasets, extending sentences to paragraphs. While searching for images by tags or sentences are of their own research interest, they were chosen in favor of visual content analysis, and are unrepresentative of real-user queries. Consequently, to what extent conclusions established on the base of these benchmarks can be generalized to real-world image retrieval remains unclear.

To promote research on web image retrieval with real-user queries, Hua *et al.* contribute Clickture-Lite, a click-through dataset down-sampled from one-year click log of the Bing image search engine [8]. This set contains one million images, 11.7 million distinct queries, and 23 million 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. Although there are a growing amount of works on exploiting Clickture-Lite [12], [21], [22], [28]–[31], they mainly focus on investigating the possibility of learning image retrieval models from the click-through data. There lacks a systematic study that reveals what kind of queries the state-of-the-art can now handle. Moreover, and somewhat surprisingly, we observe that while more sophisticated models are introduced, they are hardly compared to simple approaches that compute cross-media similarity using relatively straightforward image-to-text [32] or text-to-image mappings [33]–[35]. Another surprising fact is that while statistical significance tests are the default for text retrieval evaluation [36], such tests are invisible when assessing cross-media similarity models for web image retrieval. So the second question arises as

> *Q2. how much progress the advanced models have significantly made when compared with the matching based baselines?*

Previous works on image retrieval evaluation concentrate on content-based image retrieval (CBIR), where one searches for images by a given image [37]–[39]. By contrast, we focus on retrieving *unlabeled* images by free text. As such, evaluating cross-media similarity between an image and a textual query is essential.

In this paper, we take a pragmatic approach to answering the above two questions, and consequently make the following contributions.

- For answering Q1, we introduce *query visualness*, a quantifiable property for categorizing large-scale queries. Query visualness is found to be correlated with image retrieval performance. Such a connection helps us understand the merit and limit of the current models for addressing real-user queries.
- For answering Q2, we propose a simple text2image model, inspired by [33], [34] but redesigned to better exploit click-through data for cross-media similarity computation. We present a systematic evaluation, comparing three advanced models with simple ones on three test sets, *i.e.,* MIRFlickr [23] for single-tag queries and Clickture-dev [8] and IRC-MM15-test[1] for real-user queries. Consequently, we establish a new baseline for web image retrieval in the wild.

The techniques developed in this work have resulted in a winning entry in the Microsoft Image Retrieval Challenge at ACMMM2015 [35]. Compared to our conference paper [35] which focuses on model fusion, this work not only extends the evaluation by including more models and datasets, and consequently more experiments. More importantly, we introduce query visualness based analytics. All this results in a number of new observations and conclusions that are not covered by the conference paper. Source code is available at https://github.com/danieljf24/cmrf.

II. Related Work

This work is an endeavor towards quantifying the progress on cross-media similarity computation for web image retrieval. So we first clarify our novelty in the context of image retrieval evaluation in general. As the proposed evaluation method is at the crossroad of cross-media similarity computation and image query log analysis, we later review progress in these two fields.

A. Image Retrieval Evaluation

Previous efforts on image retrieval evaluation focus on CBIR, where a user query is represented by a specific image. In an early work [37], Shirahatti and Barnard introduce a system for making grounded comparisons of different CBIR systems, using Corel images as their dataset. Shen and Shepherd [38] propose a stratified sampling based approach for evaluating CBIR systems, providing both efficient and statistically-sound performance evaluation. Deselaers *et al.* [39] present an experimental comparison of a large number of different low-level image descriptors. As the similarity between the query image and the images being retrieved is computed directly in a visual feature space, cross-media similairty evaluation is out the scope of the above works.

To promote research on image retrieval by tags, Huiskes and Lew [40] provide MIRFlickr, a novel benchmark set collected from Flickr, followed by the NUS-WIDE dataset from Chua *et al.* [24]. Sun *et al.* [41] conduct an empirical evaluation on tag-based social image retrieval, comparing the effectiveness of multiple tag-based ranking criteria. The study by Cheng *et al.* [42] empirically investigates the effects of multiple information evidences on social image retrieval, where a query consists of a query tag and an example image to facilitate different retrieval strategies. To attack the unreliability of social image tagging, Cui *et al.* [43] introduce a supervision step into the neighbor voting scheme [44] to make the neighbors reweighted towards optimizing the ranking performance

[1] http://research.microsoft.com/en-US/projects/irc/

of tag-based image retrieval, while Cui *et al.* [45] improve neighbor voting by fusing multiple visual features. Besides tag-based image retrieval, we go a step further by considering real-user queries from a commercial web image search engine.

B. Cross-Media Similarity Computation

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

Regarding the forms, the main stream is to place an affine transformation either at the image side to project images into a bag-of-words space [46] or at the query side to project queries into a visual feature space [20], or per side a transformation to construct a latent space [11], [13], [47]. 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 [11], [48], or as Polynomial Semantic Indexing (PSI) [47] if a marginal ranking loss is minimized. In [30], 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.* [22] 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.

For the success of deep learning in computer vision and natural language processing, we observe an increasing use of such techniques as an alternative to the affine transformation. In [12], 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 query side. He *et al.* employ two CNNs for image and query embedding respectively [7]. Frome *et al.* in their DeViSE model use a pre-trained word2vec model for query embedding [9]. In a follow up work, Norouzi *et al.* employ word2vec for both query and image embedding [49]. Bai *et al.* remove the query embedding part by using a CNN that outputs a bag-of-words vector for an input image [50]. Wu *et al.* [21] employ a graph-based representation learning algorithm to incorporate implicit connections among the click-through logs, with the objective to minimize the negative log-likelihood. While the models are becoming more sophisticated, the insight into the problem appears to be limited due to the lack of a joint analysis about the retrieval performance and properties of queries. Moreover, there is hardly any comparison between these advanced models and naive ones based on simple text/image matching.

C. Image Query Log Analysis

Existing works on image query log analysis mainly focus on analyzing user search behavior [51]–[53], characterized in several aspects such as what terms are used, how are they distributed, and how many terms per query. In an early work [53], Goodrum and Spink report that compared to text retrieval, users submit relatively few terms to specify their image information needs on the web. This observation is confirmed by a more recent study by Hua *et al.* [8], reporting that around 83% queries consist of two to five words. To categorize words that ordinary users used for image description and for keyword-based image retrieval, Hollink *et al.* propose a three-level abstraction [52], i.e., nonvisual, conceptual and perceptual. These three levels correspond to information that cannot be derived from the visual content alone, information about the semantics of the image, and visual properties such as color, shape and texture. Their study suggests that people tend to use more specific terms and less abstract and perceptual terms for image retrieval than for image description. After analyzing some query log data of a local image search engine, Pu [51] finds that failed queries are longer, more distinct and unique than successful queries. These works provide good insights into image search behavior. However, query categorization is conducted by hand, and thus cannot be applied for large-scale query log analysis. Moreover, the analysis is not linked to the evaluation of image retrieval models.

III. Our Approach

A. Cross-Media Similarity Evaluation

Given an unlabeled image x and a textual query q, we aim to construct a real-valued function $f(x, q)$ that computes the cross-media similarity for the given image-query pair. To simplify the notation, x also indicates a d_i-dimensional visual feature vector extracted from the image, while q represents a d_t-dimensional bag-of-words vector extracted from the query. Apparently, the image feature x and the query feature q reside in different feature spaces, so they are not directly comparable. We need to find a common space to represent them so that cross-media similarity can be computed. Similar to previous works [12], [30], [31], [35], we also build the common space by learning from large-scale click-through data, denoted as $\mathcal{D} = \{(image, query, click)\}$.

As discussed in Section II-B, the main stream is to implement the common space via varied semantic embedding techniques. More formally, suppose that the common space has a dimensionality of d_c, with $d_c \leq \min\{d_i, d_t\}$ typically. We look for two transformations, $\phi_i(x) : \mathbb{R}^{d_i} \rightarrow \mathbb{R}^{d_c}$ and $\phi_t(q) : \mathbb{R}^{d_t} \rightarrow \mathbb{R}^{d_c}$, that embed x and q into the common space respectively. Consequently, the cross-media similarity function $f(x, q)$ can be expressed in terms of $\phi_i(x)$ and $\phi_t(q)$, say, as the form of an inner product.

Before delving into more sophisticated models for constructing $\phi_i(x)$ and $\phi_t(q)$, we first describe in Section III-A1 two baseline methods, i.e., image2text and text2image. Later in Section III-A2 we depict three representative works on semantic embedding, following the line of an increasing use of deep learning components.

1) Two Baselines: The following methods are considered as baselines, because they essentially compute cross-media similarity by straightforward image/text matching, without resorting to advanced learning techniques.

Baseline 1: image2text [32]. For a given image x, we retrieve its k nearest visual neighbors, denoted as $\{x_1, \ldots, x_k\}$, from $\mathcal{D}$. The distance between two images is computed using the Euclidean Distance between their visual feature vectors. Cross-media similarity between the image x and a given query q is computed as a weighted sum of the textual similarity between q and queries associated with each of the k neighbor images. That is,

$$f_{i2t}(x,q) := \frac{1}{k}\sum_{i=1}^{k} sim(x,x_i)\cdot sim_{i2t}(x_i,q), \quad (1)$$

where $sim(x,x')$ is an image-wise visual similarity, and

$$sim_{i2t}(x_i,q) := \frac{1}{m_i}\sum_{j=1}^{m_i} sim(q,q_{i,j})\cdot \log(click_{i,j}), \quad (2)$$

where m_i is the number of queries associated with the neighbor image x_i, $click_{i,j}$ is the click count x_i received with respect to query $q_{i,j}$, and $sim(q,q')$ is a query-wise similarity. Following [32] we use the Jaccard similarity to realize $sim(q,q')$. Viewing each query as a set of words, this similarity computes $sim(q,q')$ as $\frac{|q\cap q'|}{|q\cup q'|}$, where $|q\cap q'|$ is the number of common words between q and q', and $|q\cup q'|$ indicates the total number of unique words of the two queries. We have also experimented the cosine similarity given queries represented by tf-idf vectors, and found the Jaccard similarity still better. The contribution of query $q_{i,j}$ associated with the i-th neighbor image is weighted using its click count, see Eq. 2. As the count resides in a large range, we impose a log scale to improve stability.

Baseline 2: text2image. The text2image method reverses the mapping direction, projecting queries into the visual feature space. Given a test query q, we first retrieve the top k most similar queries, denoted as $\{q_1,\ldots,q_k\}$, from $\mathcal{D}$. The similarity between two queries is computed by the Jaccard similarity coefficient between their words, as done in the image2text method. For a web image search engine, the same query can be submitted by distinct users as time goes. Such a situation is not uncommon as there is evidence showing newly-appeared queries only contribute to a relatively small proportion (around 11%) of daily queries [54]. What an earlier user has clicked might also be relevant with respect to a later request. We leverage this intuition by setting k to be 1 if the test query can be found in the provided query log $\mathcal{D}$.

For query representation, [33] uses all images associated with the neighbor queries. This design is problematic as many of the images are irrelevant and thus noisy. Alternatively, [34] uses only the top five images most visually similar to the test image. This strategy is also questionable because the test image itself can be irrelevant to the test query. In order to represent the test query q by a set of truly relevant images, for each candidate image x_i from the j-th neighbor query q_j, we estimate the relevance score between the test query and the candidate image by jointly considering the relevance between x_i and q_j and the relevance between q_j and q, *i.e.,*

$$sim_{t2i}(x_i,q) := \log(click_{i,j})\cdot sim(q,q_j). \quad (3)$$

Accordingly, we sort all the candidate images in descending order by $sim_{t2i}(x_i,q)$, obtaining an ordered list of images $\{x_1,\ldots,x_{k'}\}$. Note that for a candidate image that is associated with multiple queries, its sim_{t2i} score is accumulated over the queries. Consequently, cross-media similarity between the image x and the query q is computed as a weighted sum of the visual similarity between x and $\{x_1,\ldots,x_{k'}\}$. That is,

$$f_{t2i}(x,q) := \frac{1}{k'}\sum_{i=1}^{k'} sim(x,x_i)\cdot sim_{t2i}(x_i,q). \quad (4)$$

To improve text matching, for both methods we conduct a standard text preprocessing: removing punctuation and lemmatizing words by the NLTK toolkit [55]. Meaningless words in the context of image retrieval like 'image' and 'picture' and standard English stopwords are also removed. The parameter k in image2text and text2image are empirically set to 50 and 30, respectively.

2) Three Semantic Embedding Models: Among the many models, we opt to implement PSI [47] and DeViSE [9], as they are key ingredients in varied methods for cross-media similarity computation. In addition, we consider ConSE [49], which is fully unsupervised and thus works across multiple datasets with ease. The influence of these works is also demonstrated by the number of citations.

Model 1: PSI. It employs two affine transformations to project both image and query into a latent common space $\mathbb{R}^{d_c}$, i.e.,

$$\begin{cases} \phi_i(x) &= W_i x \\ \phi_t(q) &= W_t q \end{cases} \quad (5)$$

where $W_i \in \mathbb{R}^{d_c\times d_i}$ and $W_t \in \mathbb{R}^{d_c\times d_t}$ are trainable matrices. The cross-media similarity is computed as a dot product between the embedding vectors,

$$f_{psi}(x,q) := (W_i x)^T (W_t q). \quad (6)$$

The two matrices are optimized by minimizing a marginal ranking loss. Concretely, we construct a large set of triplets $\mathcal{T} = \{(q,x^+,x^-)\}$ from $\mathcal{D}$, where x^+ and x^- indicate images relevant and irrelevant with respect to q. The loss function is defined as

$$L_{psi} := \sum_{(q,x^+,x^-)\in\mathcal{T}} max(0, 1 - f_{psi}(x^+,q) + f_{psi}(x^-,q)). \quad (7)$$

We minimize L_{psi} using stochastic gradient descent with a mini-batch size of 100. In addition, we use an exponentially decaying learning rate, found to be useful for large-scale optimization [56]. Since PSI requires a predefined query vocabulary, we follow [12], [48], [50], preserving up to 50k most frequent words in the training data.

Model 2: DeViSE. The main difference between PSI and DeViSE is that the latter replaces the linear transformation W_t by a pre-trained word2vec model to obtain $\phi_t(q)$. Since the training process of word2vec is highly scalable and efficient, it builds embedding vectors for millions of words with ease. Therefore, the size of the query vocabulary DeViSE can handle is much larger than in PSI.

TABLE I
MAIN PROPERTIES OF THE FIVE MODELS IMPLEMENTED IN THIS WORK.

Model	$\phi_i(x)$	$\phi_t(q)$	$f(x,q)$
image2text [32]	bag-of-words	q	Eq. (1)
text2image (*this work*)	x	visual feature	Eq. (4)
PSI [47]	$W_i x$	$W_t q$	$\phi_i(x)^T \phi_t(q)$
DeViSE [9]	$W_i x$	word2vec	$\phi_i(x)^T \phi_t(q)$
ConSE [49]	word2vec	word2vec	$cosine(\phi_i(x), \phi_t(q))$

The embedding vector of a query q is obtained by mean pooling over each word in the query:

$$\phi_t^{devise}(q) := \frac{1}{|q|} \sum_{w \in q} v(w), \tag{8}$$

where $v(w)$ corresponds to the embedding vector of each word, and $|q|$ the query length. Recent studies report that word2vec trained on many Flickr tags better captures visual relationships than its counterpart learned from web documents [57], [58]. We follow such a tactic, training a 200-dimensional word2vec model on user tags of 25M Flickr images using the skip-gram algorithm [59].

The cross-media similarity is computed as

$$f_{devise}(x,q) := (W_i x)^T \phi_t^{devise}(q). \tag{9}$$

Due to the use of word2vec, DeViSE only needs to train the image transformation matrix W_i, which is optimized in a similar way as PSI. For a fair comparison, the dimension of PSI's common space is also set to 200.

Model 3: ConSE. Compared to DeViSE, ConSE goes one step further by employing a deep visual recognition model to embed the image to the word2vec space. For a given image, a pre-trained deep model is used to predict the top $m = 10$ most relevant labels, denoted as $\{y_1, \ldots, y_m\}$. The image embedding vector is obtained as convex combination of the embedding vectors of the labels, i.e.,

$$\phi_i^{conse}(x) := \frac{1}{Z} \sum_{i=1}^{m} p(y_i|x) \cdot v(y_i) \tag{10}$$

where $p(y_i|x)$ is the relevance score of y_i given x, and $Z = \sum_{i=1}^{m} p(y_i|x)$ is a normalization factor. The cross-media similarity is computed as cosine similarity in the word2vec space,

$$f_{conse}(x,q) := cosine(\phi_i^{conse}(x), \phi_t^{devise}(q)) \tag{11}$$

In contrast to the previous two models, ConSE is fully unsupervised. Its effectiveness relies on the quality of the top predicted labels for describing the image content.

For the ease of reference, Table I presents the main properties of the two baselines and the three advanced models. For PSI, DeViSE and ConSE, the previous text preprocessing is also conducted in advance to query embedding.

3) Cross-media Similarity Fusion: Since the above methods compute cross-media similarity by distinct mechanisms, their output may complement each other. To investigate whether combining them helps, we investigate cross-media similarity fusion.

For a given image-query pair, let $\{f_i(x,q) | i = 1, \ldots, d\}$ be cross-media similarity scores computed by d distinct models. We consider the following late fusion strategy, for its simplicity and flexibility to employ a number of off-the-shelf learning to rank techniques:

$$f_\Lambda(x,q) := \sum_{i=1}^{d} \lambda_i \cdot \sigma(f_i(x,q)), \tag{12}$$

where $\Lambda = \{\lambda_i\}$ are weights to be optimized, and $\sigma(\cdot)$ is a sigmoid function for rescaling the input.

For the fusion weights, the simplest choice is to take uniform weights. Despite its simplicity, this choice often works well in practice when the similarity functions to be fused are relatively close in terms of their performance and complementary to each other. Once some ground truth data are provided, a range of learning to rank algorithms can be employed to find better weights. We utilize Coordinate Ascent [60], a greedy algorithm capable of directly optimizing (non-differentiable) performance metrics such as Average Precision and NDCG adopted in our experiments.

B. Visualness based Query Log Analysis

It is clear that not all queries can be handled by a specific image retrieval model. Knowing what kind of queries the model can address (or not) is beneficial, as it shows directions for improvement. However, due to the complexity and diversity of real user queries, devising a comprehensive query categorizing scheme is extremely difficult, if not impossible. As discussed in Section II-C, existing categorization criteria such as query uniqueness [51] and nonvisual/conceptual/perceptual [52] are subjective and cannot be numerically computed, making them inapplicable for image query log analysis at large-scale.

Intuitively, a query is more visual oriented if it contains more visual concepts. E.g., 'dog and cat' has a more clear visual correspondence than 'saying and quote'. Departing from this intuition, we propose to measure the visualness of a query by counting the proportion of its words that correspond to visual concepts. Each query, by comparing its visualness score against a given threshold, can be automatically classified as either visual or nonvisual. This classification not only helps reveal the percentage of visual oriented queries in reality, but also enables a fine-grained analysis of how a specific model responds to the two classes of queries.

Although there is no authoritative list of visual concepts, we use the 21,841 concepts from ImageNet [61], the largest labeled image collection in the public literature. Each concept in ImageNet corresponds to a specific WordNet synset, describing visual objects and scenes in the world. The concept is associated with one or more words or phrases. Since individual words in a phrase, e.g., 'hot' in 'hot dog', are not necessarily

TABLE II
SOME QUERIES AND THEIR VISUALNESS SCORES COMPUTED USING EQ. (13). QUERY WORDS FULLY MATCHED WITH SPECIFIC IMAGENET CLASSES ARE MARKED OUT VIA SQUARE BRACKETS. STRIKETHROUGH INDICATES STOPWORDS REMOVED BY PREPROCESSING.

Query	Visualness	Total click count
[flower]	1	220,747
[soccer ball]	1	25,575
[dog] ~~and~~ [cat]	1	3,423
[tattoo] design	0.500	59,854
barack obama [family]	0.333	1,001
hot weather [girl]	0.333	31
funny	0	578,659
saying ~~and~~ quote	0	3,687

visual, a query or its fragment has to be fully matched with the phrase. For a given query q, we define its visualness as

$$\text{visualness}(q) := \frac{\text{\# of query words fully matched in ImageNet}}{\text{\# of query words}} \tag{13}$$

Table II presents some queries and their visualness scores. The previous text preprocessing is applied before visualness analysis. While being precise, a drawback of the vocabulary based measure is that it cannot handle visual words outside the ImageNet vocabulary. For instance, celebrity queries such as 'barack obama' should have larger visualness scores, because of their correspondences to specific visual instances. However, they are not covered by the current ImageNet concepts. We argue that this limitation can be resolved to some extent by adding more domain-specific visual concepts. An experiment regarding celebrity queries is provided in Section IV-C. The vocabulary might be expanded automatically in a data-driven manner, by identifying new words that are suited for describing the visual content [62], [63]. A joint exploration of the vocabulary-based approach and the data-driven approach opens up possibilities for further improvement.

The dotted curve in Fig. 2 shows the percentage of visual-oriented queries in the Clickture-Lite dataset, given varied thresholds. Queries with visualness scores exceeding 0.6 are less than 25%. This result suggests that even though the number of visual concepts we can learn (from the ImageNet) is bigger than ever, they address only a relatively small part of real-user queries.

Given queries of varied visualness, we further investigate how well the image search engine tackles them. Because each query's click count reflects the chance of the search engine successfully returning relevant images for the query [64], we re-weight the query in terms of this value. Given a specific threshold, the percentage of visual-oriented queries weighted with click count is computed as

$$\frac{\sum_{q, visualness(q) > threshold} click_q}{\sum_{q'} click_{q'}},$$

where $click_q$ is the accumulated click count of query q. The updated percentage is shown as the solid curve in Fig. 2, which goes above the original curve as the threshold increases, suggesting the current search engine better handles visual-oriented queries. This result also conforms to our daily experience with web image search.

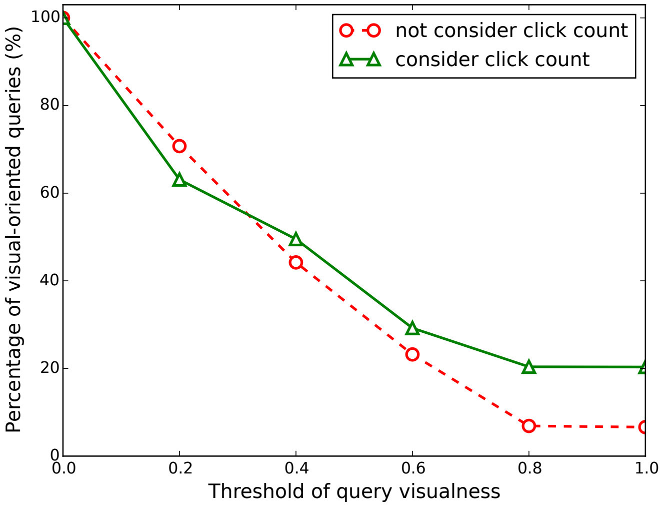


Fig. 2. **Percentage of visual-oriented queries in the Clickture-Lite dataset**. A query is classified as visual-oriented if its visualness score exceeds a given threshold. Queries of larger visualness scores receive more user clicks, indicating that they are better answered by the current image search engine.

IV. EVALUATION

To evaluate the effectiveness of the five models for answering queries of varied difficulty, we perform image retrieval by single-tag queries and by real-user queries. We choose MIRFlickr-test, a popular benchmark set for tag-based image retrieval [23]. Popularized by the Microsoft Image Retrieval Challenge [8], Clickture-dev is becoming a benchmark set for web image retrieval by real-user queries [12], [21], [22], [30], [50]. Nonetheless, the public availability of both images and ground-truth labels may unconsciously increase the chance of overfitting. Hence, we further include in our evaluation the test set from the Microsoft Image Retrieval Challenge at ACMMM'2015, which we term IRC-MM15-test. An overview of the three test sets is given in Table III. More details will be given in the following experiments.

Evaluation criteria. As Mean Average Precision (MAP) is commonly used on MIRFlickr [4], we follow this convention. For Clickture-dev and IRC-MM15-test, we follow the Image Retrieval Challenge protocol [8], reporting Normalized Discounted Cumulated Gain (NDCG) at the rank of 25, *i.e.,*

$$NDCG_{25} = 0.01757 \sum_{i=1}^{25} \frac{2^{rel_i - 1}}{log_2(i+1)}, \tag{14}$$

where $rel_i = \{Excellent = 3, Good = 2, Bad = 0\}$ is the ground truth label of the i-th ranked image with respect to the query. The constant 0.01757 is a normalization factor to ensure that an ideal ranking will have an NDCG score of 1. The overall performance is measured by averaging NDCG scores over the test queries. As a sanity check, we report the performance of a random baseline on each test set, obtained by sorting images in terms of scores generated at random. On MIRFlickr-test, the random baseline has MAP of 0.0720,

TABLE III
THREE TEST SETS USED IN OUR EXPERIMENTS. MODELS EVALUATED ON MIRFLICKR-TEST ARE LEARNED FROM THE MIRFLICKR TRAINING SET [23], WHILE MODELS EVALUATED ON CLICTURE-DEV AND IRC-MM15TEST ARE LEARNED FROM THE CLICKTURE-LITE DATASET [8].

Test set	**Source**	**Queries**	**Images**	**Image-query pairs**
MIRFlickr-test	Flickr	14 singel-tag queries	10,000	140,000
Clickture-dev	Web	1,000 real-user queries	79,665	79,926
IRC-MM15test	Web	9,949 real-user queries	147,346	579,122

and NDCG_{25} of 0.4702 and 0.4260 on Clickture-dev and IRC-MM15-test.

Test of statistical significance. We conduct a randomization test [36], with the null hypothesis that there is no performance difference in two image retrieval systems. Given a set of test queries and two retrieval systems, A and B, which have been evaluated per query. Let $diff$ be the absolute difference between the averaged performance scores of the two systems. To check if the difference is caused by chance, for half of the test queries that are selected at random, their performance scores are switched between A and B. The absolute performance difference between A and B is re-computed accordingly and compared against $diff$. The trial is repeated n times. The p-value of the randomization test is defined as the percentage of the trials having the new difference larger than $diff$. As suggested in [36], one can comfortably reject the null hypothesis if the p-value is smaller than 0.05. In other words, the performance difference between two image retrieval systems is considered statistically significant if the p-value produced by the randomization test is smaller than 0.05.

A. Experiment 1. Single-tag Queries

Dataset. The MIRFlickr set [23] contains 25,000 Flickr images with ground truth annotations available for 14 tags such as 'car', 'dog' and 'river'. Following the official participation, we use 10,000 images as the test set, termed as MIRFlickr-test. The remaining 15,000 images are used for model training. As all the test images are labeled with respect to the 14 test queries, this results in 14×10,000=140,000 image-query pairs for MIRFlickr-test.

Visual features. For visual feature extraction, we employ three pre-trained CNN models, i.e., CaffeNet [65], VggNet [66], and GoogleNet [67], as their distinct network architectures may yield complementary visual features. They were learned from examples of 1,000 ImageNet classes defined in the Large Scale Visual Recognition Challenge [68]. For CaffeNet and VggNet, we use the last fully connected layer, resulting in visual feature vectors of 4,096 dimensions. For GoogleNet, we use its pool5 layer, but replace the default 7×7 filter by a 4×4 filter to better capture spatial layouts. This also results in a feature vector of 4,096 dimensions. Note the three CNN models are separately used as the visual recognition component in ConSE. For simplicity, the visual features are named after the corresponding CNN models. CaffeNet is the default feature unless stated otherwise.

TABLE IV
PERFORMANCE OF DIFFERENT MODELS ON ANSWERING SINGLE-TAG QUERIES ON MIRFLICKR-TEST. AVERAGE WEIGHTS ARE USED FOR BOTH FEATURE-FUSION AND METHOD-FUSION.

Method	CaffeNet	VggNet	GoogleNet	*Feature-fusion*
random	0.0720	0.0720	0.0720	-
Upper bound	1.0	1.0	1.0	-
image2text	0.4416	0.4895	0.4527	0.5363
text2image	0.4414	0.4930	0.4753	0.5226
PSI	0.4689	0.5358	0.5323	**0.6544**
DeViSE	0.4626	0.5342	0.5036	0.6312
ConSE	0.3419	0.3750	0.3745	0.4370
Method-fusion	0.5643	**0.6383**	0.5972	0.6655

Advanced models versus baselines. The performance of different methods on MIRFlickr-test is summarized in Table IV. All the methods are noticeably better than the random result. Among them, PSI is at the leading position, followed by DeViSE. Recall that their main difference between lies in the underlying approach to query embedding. The result suggests that for embedding single-tag queries, a task-specific transformation matrix is more suited than a word2vec model learned in advance.

The above conclusion is further supported by the relatively lower performance of ConSE, which fully uses pre-trained models for both query and image embedding. While sharing the same word2vec model with DeViSE, at the image side ConSE counts on the 1K ImageNet classes to describe the visual content, which are too specific to represent the 14 test queries. Moreover, the advantage of one model over another is feature independent. For instance, PSI consistently outperforms DeViSE given all the three features.

Concerning the two baseline models, image2text with feature-fusion scores higher MAP of 0.5363 than text2image with MAP of 0.5226. Nonetheless, with p-value of 0.775 by randomization test, the difference is not statistically significant.

The influence of fusion. As shown in Table IV, average-fusion, either along the line of methods (*method-fusion*) or along the line of features (*feature-fusion*), leads to significant improvements. For instance, among all the fifteen combinations of methods and features, PSI + VggNet is the best. This run can be further improved by adding PSI + CaffeNet and PSI + GoogleNet, lifting MAP from 0.5358 to 0.6544. A per-query comparison is given in Fig. 3, where fusion performs the best for the majority of the queries. Average fusion of all the fifteen combinations generates MAP of 0.6655. Moreover, we employ Coordinate Ascent to optimize the weights on a set of 5k images sampled at random from the MIRFlickr training set. With the learned weights, the performance can be further improved, reaching MAP of 0.6772.

B. Experiment 2. Real-user Queries

Test set 1: Clickture-dev. This test set contains 1,000 real-user queries and 79,665 images. Though in theory there shall be 1000×79,665 query-image pairs, only ground truth of 79,926 pairs are publicly available. Thus, image retrieval on Clickture-dev is to score and rank a subset of the images

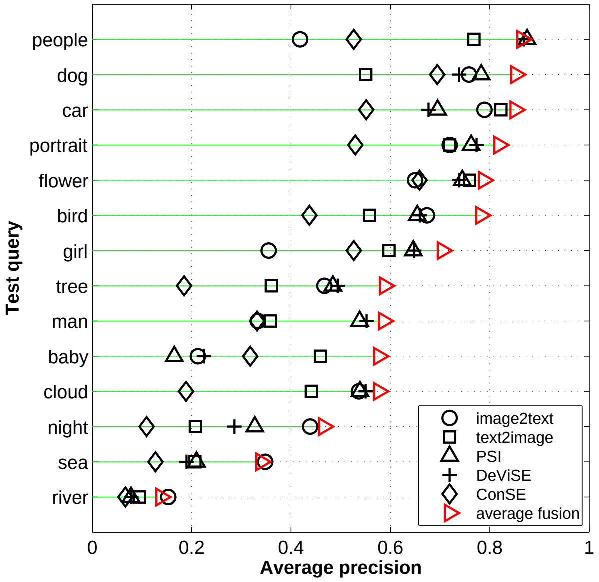


Fig. 3. **Per-query comparison on MIRFlickr-test**. Visual feature: VggNet. Queries are sorted in terms of the performance of average-fusion.

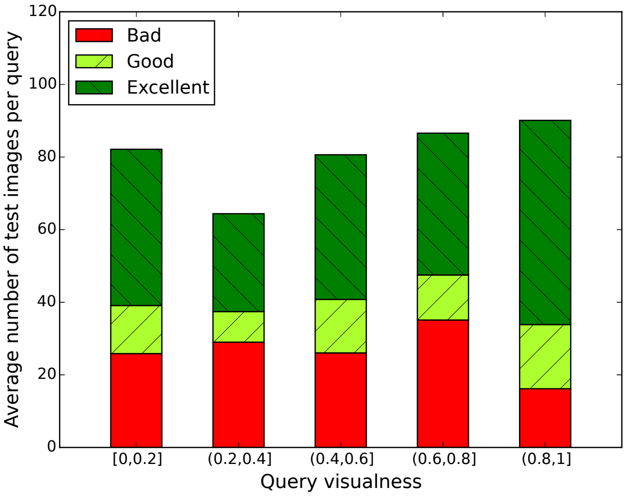


Fig. 4. **Average number of test images per query in Clickture-dev**. Queries have been grouped in terms of their visualness scores.

specific for each query. Each pair is manually rated as *Excellent*, *Good* or *Bad*, based on the relevance between image and query.The distribution of each label is visualized in Fig. 4.

Test set 2: IRC-MM15-test. This test set contains 9,949 real queries and 579,122 query-image pairs. The set is much larger and more challenging than Clickture-dev. As its ground truth is non-public, we submit our results to the task organizers and get performance scores back.

When tested on Clickture-dev and IRC-MM15-test, all the models use Clickture-Lite as the training data.

Advanced models versus baselines. Table V shows the performance of the individual methods with varied features on Clickture-dev. Note that the performance upper bound is less than 1, because 609 of the 1,000 test queries have less than 25 images labeled as Excellent. Again, all the methods beat the random result. It is worth pointing out that retrieving images from an unconstrained collection for thousands of real-user queries is a grand challenge. Hence, although the performance divergence may appear to be relatively small (see also the performance reported in [12], [21]), the significance of the individual models shall not be underestimated.

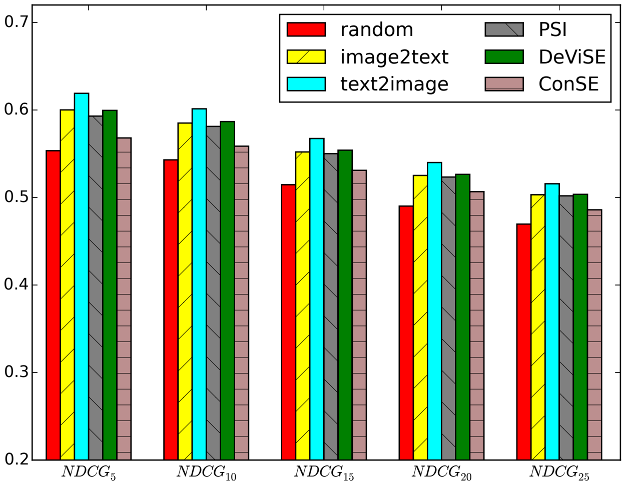


Fig. 5. **NDCG results at multiple ranks**. Test set: Clickture-dev. The text2image model performs the best.

When ranking the methods in terms of their performance, the resultant order differs much from that in the first experiment. In contrast to the scenario of single-tag queries, the second baseline method, text2image, now takes the leading position, followed by DeViSE, PSI, image2text and ConSE. Fig. 5 shows NDCG results at multiple ranks, and text2image still performs the best.

In spite of its simplicity, text2image benefits from the large-scale query log data. We find that for 42% of the test queries, there exist precise matches in Clickture-Lite. In these cases, cross-media similarity computation boils down to comparing a test image with several highly clicked images of the query. As a dual form of text2image, image2text also compares images, but works in a reverse order by finding similar images first. This may incorrectly introduce irrelevant images and consequently propagate irrelevant words to the test images. While such noisy words are not critical for single-tag queries (as in Experiment 1), they affect complex queries. Consequently, we see from Table V that text2image consistently outperforms image2text for real-user queries.

As for the semantic embedding models, in essence they aim to describe both images and queries by some latent topics. While the topics provide higher level abstraction than visual features and bag-of-words features, and this is a wanted property for visual concept search, the discrimination ability of both image and query is inevitably reduced. See Fig. 6 for instance. On the other hand, by looking into the individual queries of Clickture-dev, we observe that many of the real-user queries are related to finding instances instead of categories, as exemplified in Table VI. Consider the query ‘ling simpson’ for instance. While the true answer is about a female cartoon character, DeViSE retrieves images of real females. Again, when coming to categorical queries such as ‘family photo’ and

TABLE V
PERFORMANCE OF DIFFERENT MODELS FOR ANSWERING REAL-USER QUERIES ON CLICKTURE-DEV. AVERAGE WEIGHTS ARE USED FOR BOTH FEATURE-FUSION AND METHOD-FUSION.

Method	CaffeNet	VggNet	GoogleNet	*Feature-fusion*
random	0.4702	0.4702	0.4702	-
Upper bound	0.6842	0.6842	0.6842	-
image2text	0.5032	0.5025	0.5043	0.5055
text2image	0.5153	0.5105	0.5127	**0.5149**
PSI	0.5016	0.5028	0.5037	0.5086
DeViSE	0.5033	0.5020	0.5070	0.5099
ConSE	0.4861	0.4878	0.4837	0.4882
Method-fusion	0.5137	**0.5145**	0.5142	0.5177

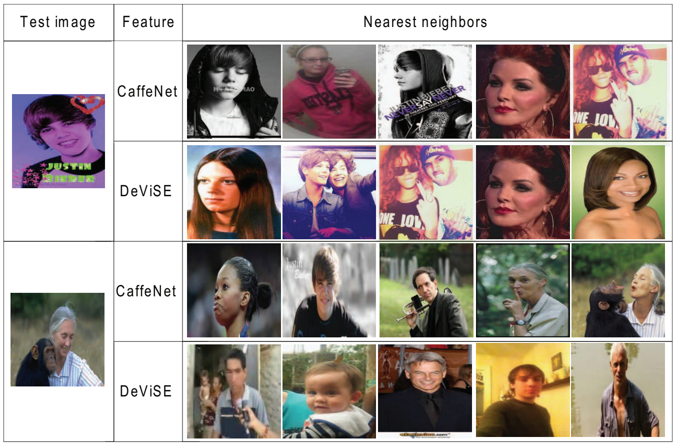


Fig. 6. **Top five similar images retrieved using the original CaffeNet feature and the subspace feature by DeViSE, separately**. The former gets more images of the same person (Justin Bierber and Jane Goodall) as in the test images.

'woman bicycle', DeViSE successfully find relevant images. Therefore, despite their superior performance for single-tag queries which are visual concepts, DeViSE and PSI are less effective for real-user queries.

The influence of fusion. Similar to Experiment 1, we investigate method-fusion and feature-fusion. As shown in Table V, in general fusion gains some performance improvement. Nonetheless, the difference between the single best run (text2image + CaffeNet, MAP of 0.5153), and average-fusion (MAP of 0.5177) does not pass the significance test. This again differs from the results of Experiment 1, where fusion brings in clear improvement.

Robust analysis. As already shown in Fig. 4, for most queries more than half of the test images are Excellent or Good, meaning a random sort may return relevant test images with good chance. We analyze the robustness of each method by adding extra noise. Concretely, for each query with n test images, we add h-fold noise, namely $h \times n$ images randomly taken from other queries, with $h = 1, 2, \ldots, 10$. The performance curves with respect to the level of noise are shown in Fig. 7. Unsurprisingly, for all methods the performance goes down. Yet, the curve of the random run drops more sharply than the others, indicating a more challenging start point. Given the varied levels of noise, our conclusion that the baselines are better than the advanced models still holds. Interestingly, the influence of fusion is more evident now. The result suggests that fusion robustifies cross-media similarity computation.

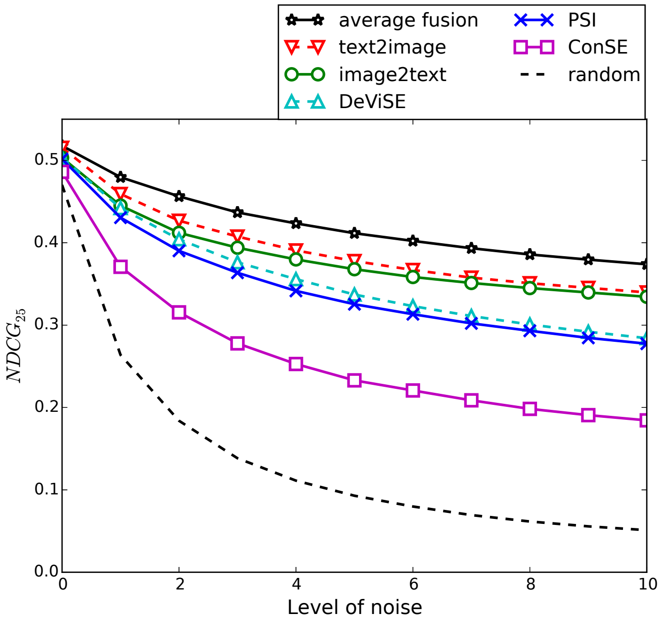


Fig. 7. **Performance curves of different methods with respect to the artificial noise**. Test set: Clickture-dev. The two baseline methods outperform the three advanced methods given a specific level of noise.

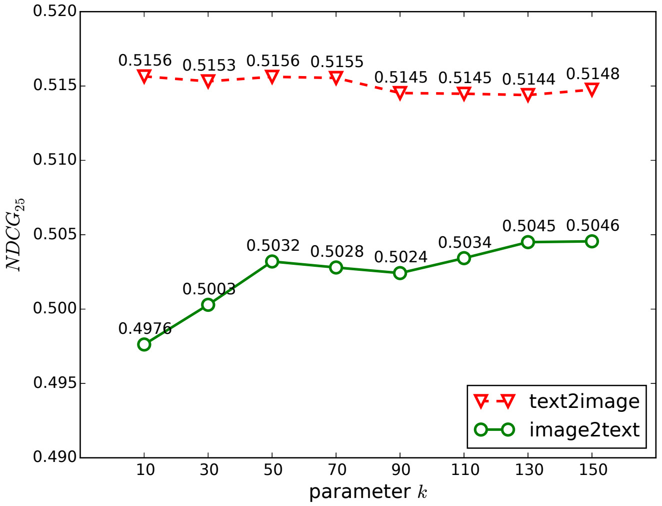
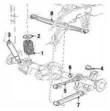
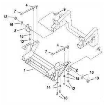
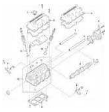
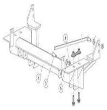
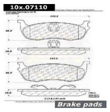


Fig. 8. **The influence of the parameter k on the two baseline methods**, showing that image2text benefits more from optimizing k. Test set: Clickture-dev.

Previously we have empirically set the parameter k, which corresponds to the number of neighbor images for image2text and the number of neighbor queries for text2image, to be 50 and 30, respectively. To reveal the influence of this parameter on the two baselines, we try k with its value ranging from 10 to 150 with an interval of 20. As shown in Fig. 8, image2text benefits more from optimizing k.

Efficiency analysis. We implement all the five models in

TABLE VI
Retrieval results by text2image and DeViSE. Excellent, Good or Bad images are indicated by ★★, ★☆, and ☆☆, respectively.

Test query	Method	NDCG@25	Queries retrieval from Clickture-Lite	Images retrieval from Clickture-dev				
2001 ford expedition part visualness: 0	text2image	0.3597	2001 ford expedition motor part ford expedition part 2001 ford expedition ford expedition part diagram ford expedition part diagram	 ★☆	 ★☆	 ☆☆	 ★☆	 ★☆
	DeViSE	0.0	2002 ford expedition part 2003 ford expedition part 1999 ford expedition part ford expedition 1999 part 1997 ford expedition part	 ☆☆	 ☆☆	 ☆☆	 ☆☆	 ☆☆
6v [battery] small visualness: 0.333	text2image	0.6206	6v battery small battery heater 6v lantern battery inside 6v battery small battery candle	 ★★	 ★★	 ★★	 ★★	 ☆☆
	DeViSE	0.2155	6v battery inside 6v battery small battery chargeing device 6volt battery labled battery	 ☆☆	 ☆☆	 ★★	 ★★	 ☆☆
[ling] simpson visualness: 0.5	text2image	0.6204	simpson ling jessica simpson cody simpson homer simpson	 ★★	 ★★	 ★☆	 ★★	 ☆☆
	DeViSE	0.1828	lisa ling bridsmains ling hairstleys ling freshwat ling ling xiaoyu	 ★☆	 ☆☆	 ☆☆	 ☆☆	 ☆☆
[family] ~~photo~~ visualness: 1	text2image	0.2059	family family family family tree family guy family quote	 ☆☆	 ☆☆	 ☆☆	 ☆☆	 ☆☆
	DeViSE	0.9278	family turs family tuohy family duggar family santorums family	 ★★	 ★★	 ★★	 ★★	 ★★
[woman] [bicycle] visualness: 1	text2image	0.7680	woman bicycle bicycle woman woman riding bicycle bicycle sizing woman trek woman bicycle	 ☆☆	 ★★	 ★★	 ★★	 ★★
	DeViSE	0.9169	woman bicycle bicycle woman bike woman woman bike bike woman bewach crusiers	 ★★	 ★★	 ★★	 ★★	 ★★

python, and employ the theano deep learning library for PSI and DeViSE. Additionally, for image2text we employ the production quantization algorithm [69] to accelerate k visual neighbor search. Clickture-Lite is used as the training data and Clickture-dev as the test data. As visual feature extraction is required by all the methods and can be precomputed, we exclude this part from comparison. Table VII provides computational cost and memory requirements of these methods on a regular PC with 32G RAM and a GTX TITANX X GPU. The trained DeViSE model is more compact and faster than text2image.

Performance on IRC-MM15-test. As aforementioned, the possibility of overfitting on Clickture-dev exists because of its full availability. Hence, we further evaluate on IRC-MM15-

TABLE VII
COMPUTATIONAL COST AND MEMORY REQUIREMENTS OF THE FIVE MODELS. TRAINING SET: CLICKTURE-LITE. TEST SET: CLICTURE-DEV. VISUAL FEATURE: CAFFENET. CONSE REQUIRES NO TRAINING AND IS THE MOST EFFICIENT FOR CROSS-MEDIA SIMILARITY COMPUTATION.

	Training		Test	
Method	Time	Memory	Time	Memory
image2text	-	-	4.4 hours	3,800 M
text2image	-	-	980 seconds	4,500 M
PSI	120 hours	1450 M	550 seconds	750 M
DeViSE	50 hours	1400 M	385 seconds	670 M
ConSE	-	-	110 seconds	400 M

TABLE VIII
PERFORMANCE ON IRC-MM15-TEST.

Method	NDCG$_{25}$
Random baseline	0.4260
Upper bound	0.6924
Top performer [35]	0.4929
This work:	
DeViSE	0.4842
text2image	0.4902
average-fusion	0.4946
learned-fusion	**0.4963**

test, the ground truth of which is unavailable to us. Table VIII presents the performance of the selected methods. The weights of learned-fusion are optimized on Clickture-dev. The result again confirms our finding that text2image surpasses DeViSE for answering real-user queries. The difference between the two is statistically significant. In addition, the best run also outperforms our conference version [35] in post-competition evaluation[2].

C. Experiment 3. Analytics using Query Visualness

Thus far all the comparisons are holistic. To gain a further understanding of the individual methods, we leverage query visualness developed in Section III-B. The 1,000 test queries from Clickture-dev are grouped according to their visualness, with the performance of each group shown in Fig. 9(a). DeViSE outperforms the two baselines for queries with visualness scores over 0.8. The result is in line with what we have observed in the visual concept search experiment.

Still, Fig. 9(a) does not allow us to conclude if visual-oriented queries are better handled, because the random run already gives relatively high NDCG of 0.6118. So we add one-fold noise to make the random runs more balanced across different groups. Observing Fig. 9(b) from left to right, the averaged gain of the five models over the random run increases along with query visualness, showing the current models better address visual-oriented queries. Moreover, since the notion of query visualness is orthogonal to the development of the cross-media models, its connection to the model performance indicates that such a query categorization is meaningful.

To further verify the necessity of the proposed query visualness measure, we check if a similar connection can be found with query length, a property frequently discussed in query log analysis. To that end, we employ Spearmman's rank correlation, which provides a nonparametric measure of monotonic relationship between two ranked variables. The correlation is computed by accumulating squared difference in paired ranks, so a perfect correlation of +1 or 1 occurs if each of the variables is a perfect monotone function of the other. In our context, the variables are the 1,000 test queries from Clickture-dev, with their ranks obtained by sorting in terms of three criteria separately, *i.e.,* performance of DeViSE, query visualness, and reciprocal of query length. Note that query length tends to be negatively correlated with the performance, so we use its reciprocal for the ease of comparison. Consequently, we compute the Spearmman correlation between the first and the second ranked lists and between the first and the third ranked lists. As shown in Fig. 10, given zero noise the reciprocal of query length, with coefficient of 0.224, appears to be better correlated to the performance when compared to query visualness with coefficient of 0.057. Looking into Clickture-dev, we find that the test images of shorter queries, in particular, with one or two words, contain many more Excellent examples. So shorter queries have better performance a priori. However, as more noise is added, the influence of such a bias is lessened. This explains why the reciprocal of query length has larger correlations at the beginning, but is surpassed by query visualness later. As the task becomes more difficult, query visualness exhibits larger monotonic correlation to the performance.

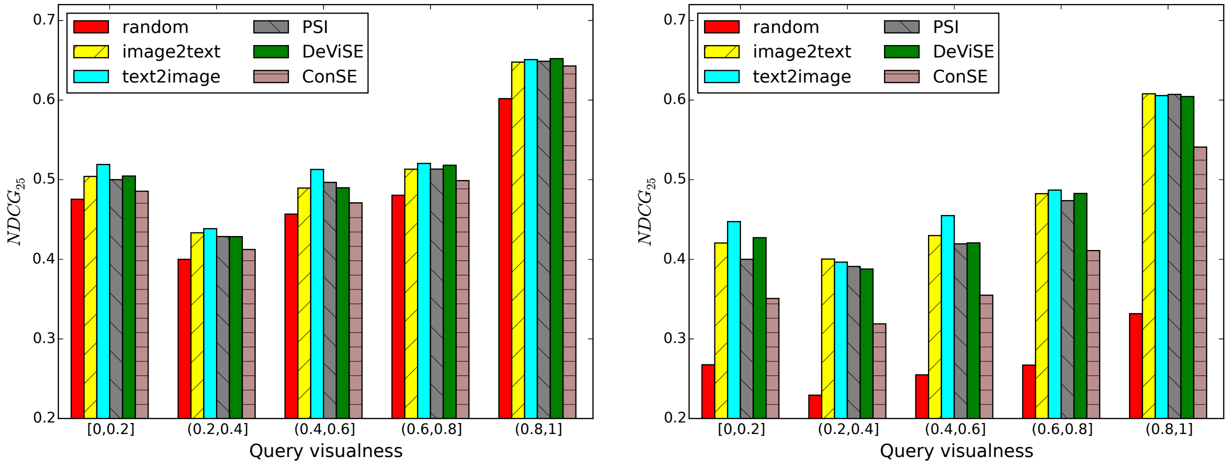
As we have mentioned in Section III-B, celebrity-related queries receive low visualness scores due to the limit of our visual concept vocabulary. Inspired by [28], we identify these queries in Clickture-dev by a semi-automatic approach as follows. Using a list of 2,657 celebrities from the Internet[3], we first build a name vocabulary by putting their first and last names together. Accordingly, we obtain over 400 test queries having at least one word from the vocabulary. A list of 240 celerity-related queries is compiled after manual verification. Performance of these queries is given in Table IX, where text2image again performs the best.

D. Experiment 4. Comparison to State-of-the-Art

Given that we have evaluated only three semantic embedding models, it would be bold to claim text2image as a new baseline. So in this part, we compare with a number of state-of-the-art works that report their performance on Clickture-dev, and thus the numbers are directly comparable. The works are:
1) CCA [48]: Find the transformation matrices that maximizes the correlation between embedding vectors of relevant image-query pairs.
2) CCL [30]: Learn a latent space by minimizing distance of relevant image-query pairs in the new space, while preserving

[2] The best run of [35] obtains NDCG$_{25}$ of 0.5200, using a search result re-ranking trick. Adding the same trick to our solution has scored NDCG$_{25}$ of 0.5312. However, this trick does not work when given individual image-query pairs, so we exclude it from comparison.

[3] http://www.posh24.com/celebrities/a_to_z



(a) Clickture-dev

(b) Adding one-fold noise to Clickture-dev

Fig. 9. **Performance of different models on (a) Clickture-dev and (b) Clickture-dev with one-fold noise**. Queries are grouped according to their visualness. The minimal performance is for query visualness in (0.2, 0.4) due to the fact that this group has the lowest percentage of relevant images, see Fig. 4, and thus image retrieval for this group is more challenging. Observing (b) from left to right, the averaged gain of the five models over the random run increases along with query visualness, from 0.1416, 0.1495, 0.1611 to 0.2003 and 0.2616, showing the current models better address visual-oriented queries.

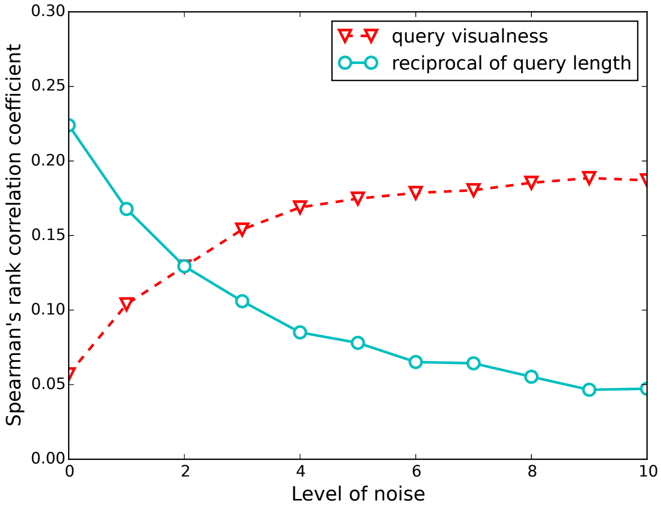


Fig. 10. **Spearman's rank correlation coefficients between query visualness / reciprocal of query length and the performance of DeViSE, under different levels of noise**. Test set: Clickture-dev. As the task becomes more difficult, query visualness exhibits stronger monotonic correlation to the performance.

TABLE IX
PERFORMANCE OF CELEBRITY-RELATED QUERIES ON CLICKTURE-DEV. METHODS SORTED IN TERMS OF THEIR NDCG_{25} SCORES.

Method	NDCG_{25}
random	0.4423
ConSE	0.4441
image2text	0.4638
DeViSE	0.4611
PSI	0.4655
text2image	**0.4944**

the structure in the original feature space.
3) CSM [12]: Project images and queries into a latent space by a deep CNN and an affine transformation, respectively.
4) BoWDNN [50]: Extract a bag-of-words vector from an input image using a deep CNN.
5) MRW-NN [21]: A graph-based representation learning algorithm to generate a common space wherein images and queries strongly connected in click-through logs are close.
6) RCCA [22]: 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.

The performance of the above works is listed in Table X. Our methods produce larger NDCG_{25} scores. Due to the lack of per-query scores, we are unable to conclude if the difference is significant. Nevertheless, we empirically find out that when the performance difference between two image retrieval systems, *i.e.,* the $diff$ to be compared in the randomization test, is larger than 0.005, it is often sufficient to pass the significance test. Hence, our fusion results are likely to be significantly better than the state-of-the-art.

We compare the proposed text2image method with two alternatives, namely query-based scoring [33] which uses all images from the neighbor queries and online classification [34] which uses only the most similar images to a test image. Given the same CaffeNet feature, our method with NDCG_{25} of 0.5153 is significantly better than the two alternatives, scoring NDCG_{25} of 0.4905 and 0.4958, respectively.

V. Summary and Conclusions

As an initial effort to quantify progress on web image retrieval, this paper presents a systematic study that combines large-scale query log analysis and state-of-the-art evaluation of cross-media similarity models.

Conclusions we can offer are as follows. Given the proposed text2image method as the baseline, much progress has already

TABLE X
COMPARING WITH THE-STATE-OF-ART ON CLICKTURE-DEV. THE PROPOSED TEXT2IMAGE IS ON PAR WITH THE STATE-OF-THE-ART, AND CAN BE FURTHER IMPROVED BY SIMPLE AVERAGE-FUSION.

Method	NDCG_{25}
CCA [48]	0.5055
CCL [30]	0.5059
CSM [12]	0.5070
BoWDNN [50]	0.5089
MRW-NN [21]	0.5104
RCCA [22]	0.5112
This work:	
text2image	0.5153
average-fusion	**0.5177**

been made by the advanced semantic embedding models, but the progress is mainly attributed to their relatively good performance on visual-oriented queries. However, this class of queries accounts for only a small part of real-user queries. Image retrieval experiments on the Clickture dataset shows that text2image outperforms several recently developed deep learning models including DeViSE [9], ConSE [49], BoWDNN [50], MRW-NN [21], and RCCA [22]. For web image retrieval in the wild, we recommend text2image as a new baseline to be compared against when one advocates novel cross-media similarity models.

ACKNOWLEDGMENT

This work was supported by National Science Foundation of China (No. 61672523).

REFERENCES

[1] P. Enser, “The evolution of visual information retrieval,” *J. Inf. Sci.*, vol. 34, no. 4, pp. 531–546, 2008.

[2] Y. Yang, Y. Zhuang, F. Wu, and Y.-H. Pan, “Harmonizing hierarchical manifolds for multimedia document semantics understanding and cross-media retrieval,” *IEEE Trans. Multimedia*, vol. 10, no. 3, pp. 437–446, 2008.

[3] C. Kang, S. Xiang, S. Liao, and C. Xu, “Learning consistent feature representation for cross-modal multimedia retrieval,” *IEEE Trans. Multimedia*, vol. 17, no. 3, pp. 370–381, 2015.

[4] X. Li, T. Uricchio, L. Ballan, M. Bertini, C. Snoek, and A. Del Bimbo, “Socializing the semantic gap: A comparative survey on image tag assignment, refinement and retrieval,” *ACM Computing Surveys*, vol. 49, no. 1, pp. 14:1–14:39, 2016.

[5] Y. T. Zhuang, Y. Yang, and F. Wu, “Mining semantic correlation of heterogeneous multimedia data for cross-media retrieval,” *IEEE Trans. Multimedia*, vol. 10, no. 2, pp. 221–229, 2008.

[6] C. Kofler, M. Larson, and A. Hanjalic, “Intent-aware video search result optimization,” *IEEE Trans. Multimedia*, vol. 16, no. 5, pp. 1421–1433, 2014.

[7] Y. He, S. Xiang, C. Kang, and J. Wang, “Cross-modal retrieval via deep and bidirectional representation learning,” *IEEE Trans. Multimedia*, vol. 18, no. 7, pp. 1363–1377, 2016.

[8] 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 Multimedia*, 2013, pp. 243–252.

[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, pp. 2121–2129.

[10] Y. Hua, S. Wang, S. Liu, A. Cai, and Q. Huang, “Cross-modal correlation learning by adaptive hierarchical semantic aggregation,” *IEEE Trans. Multimedia*, vol. 18, no. 6, pp. 1201–1216, 2016.

[11] 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.*, vol. 36, no. 3, pp. 521–535, 2014.

[12] 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*, vol. 17, no. 11, pp. 2000–2007, 2015.

[13] C. Deng, X. Tang, J. Yan, and W. Liu, “Discriminative dictionary learning with common label alignment for cross-modal retrieval,” *IEEE Trans. Multimedia*, vol. 18, no. 2, pp. 208–218, 2016.

[14] Y. Gong, Q. Ke, M. Isard, and S. Lazebnik, “A multi-view embedding space for modeling internet images, tags, and their semantics,” *IJCV*, vol. 106, no. 2, pp. 210–233, 2014.

[15] X. Li, C. G. M. Snoek, and M. Worring, “Learning social tag relevance by neighbor voting,” *IEEE Trans. Multimedia*, vol. 11, no. 7, pp. 1310–1322, 2009.

[16] M. Wang, K. Yang, X. S. Hua, and H. J. Zhang, “Towards a relevant and diverse search of social images,” *IEEE Trans. Multimedia*, vol. 12, no. 8, pp. 829–842, 2010.

[17] X. Ding, B. Li, W. Xiong, and W. Guo, “Multi-instance multi-label learning combining hierarchical context and its application to image annotation,” *IEEE Trans. Multimedia*, vol. 18, no. 8, pp. 1616–1627, 2016.

[18] D. Lu, X. Liu, and X. Qian, “Tag-based image search by social re-ranking,” *IEEE Trans. Multimedia*, vol. 18, no. 8, pp. 1628–1639, 2016.

[19] Y. Wu, J.-Y. Bouguet, A. Nefian, and I. V. Kozintsev, “Learning concept templates from web images to query personal image databases,” in *Proc. of ICME*, 2007.

[20] Y. Liu, D. Xu, I. W.-H. Tsang, and J. Luo, “Textual query of personal photos facilitated by large-scale web data,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 1022–1036, 2011.

[21] F. Wu, X. Lu, J. Song, S. Yan, Z. M. Zhang, Y. Rui, and Y. Zhuang, “Learning of multimodal representations with random walks on the click graph,” *IEEE Trans. Image Processing*, vol. 25, no. 2, pp. 630–642, 2016.

[22] T. Yao, T. Mei, and C.-W. Ngo, “Learning query and image similarities with ranking canonical correlation analysis,” in *Proc. of ICCV*, 2015, pp. 955–958.

[23] M. J. Huiskes, B. Thomee, and M. S. Lew, “New trends and ideas in visual concept detection: the MIR flickr retrieval evaluation initiative,” in *Proc. of ACM MIR*, 2010, pp. 527–536.

[24] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. Zheng, “NUS-WIDE: A real-world web image database from National University of Singapore,” in *Proc. of CIVR*, 2009, pp. 527–536.

[25] M. Hodosh, P. Young, and J. Hockenmaier, “Framing image description as a ranking task: Data, models and evaluation metrics,” *Journal of Artificial Intelligence Research*, pp. 853–899, 2013.

[26] T. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. Zitnick, “Microsoft COCO: common objects in context,” *CoRR*, vol. abs/1405.0312, 2014.

[27] N. Rasiwasia, J. Costa Pereira, E. Coviello, G. Doyle, G. R. Lanckriet, R. Levy, and N. Vasconcelos, “A new approach to cross-modal multimedia retrieval,” in *Proc. of ACM Multimedia*, 2010, pp. 251–260.

[28] C.-C. Wu, K.-Y. Chu, Y.-H. Kuo, Y.-Y. Chen, W.-Y. Lee, and W. H. Hsu, “Search-based relevance association with auxiliary contextual cues,” in *Proc. of ACM Multimedia*, 2013, pp. 393–396.

[29] Z. Xu, Y. Yang, A. Kassim, and S. Yan, “Cross-media relevance mining for evaluating text-based image search engine,” in *Proc. of ICME*, 2014, pp. 1–4.

[30] 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, pp. 717–726.

[31] Y. Liu, Z. Shi, X. L. Li, and G. Wang, “Click-through-based deep visual-semantic embedding for image search,” in *Proc. of ACM Multimedia*, 2015, pp. 955–958.

[32] Y. Pan, T. Yao, K. Yang, H. Li, C.-W. Ngo, J. Wang, and T. Mei, “Image search by graph-based label propagation with image representation from dnn,” in *Proc. of ACM Multimedia*, 2013, pp. 397–400.

[33] Q. Fang, H. Xu, R. Wang, S. Qian, T. Wang, J. Sang, and C. Xu, “Towards msr-bing challenge: Ensemble of diverse models for image retrieval,” in *MSR-Bing IRC 2013 Workshop*, 2013.

[34] L. Wang, S. Cen, H. Bai, C. Huang, N. Zhao, B. Liu, Y. Feng, and Y. Dong, “France Telecom Orange labs (Beijing) at MSR-Bing challenge on image retrieval 2013,” in *MSR-Bing IRC 2013 Workshop*, 2013.

[35] J. Dong, X. Li, S. Liao, J. Xu, D. Xu, and X. Du, “Image retrieval by cross-media relevance fusion,” in *Proc. of ACM Multimedia*, 2015, pp. 173–176.

[36] M. Smucker, J. Allan, and B. Carterette, "A comparison of statistical significance tests for information retrieval evaluation," in *Proc. of CIKM*, 2007, pp. 623–632.

[37] N. V. Shirahatti and K. Barnard, "Evaluating image retrieval," in *Proc. of CVPR*, 2005, pp. 955–961.

[38] J. Shen and J. Shepherd, "Efficient benchmarking of content-based image retrieval via resampling," in *Proc. of ACM Multimedia*, 2006, pp. 569–578.

[39] T. Deselaers, D. Keysers, and H. Ney, "Features for image retrieval: an experimental comparison," *Information Retrieval*, vol. 11, no. 2, pp. 77–107, 2008.

[40] M. J. Huiskes and M. S. Lew, "The MIR Flickr retrieval evaluation," in *Proc. of ACM MIR*, 2008.

[41] A. Sun, S. S. Bhowmick, K. T. Nam Nguyen, and G. Bai, "Tag-based social image retrieval: An empirical evaluation," *J. Am. Soc. Inf. Sci. Technol.*, vol. 62, no. 12, pp. 2364–2381, 2011.

[42] Z. Cheng, J. Shen, and H. Miao, "The effects of multiple query evidences on social image retrieval," *Multimedia Syst.*, vol. 22, no. 4, pp. 509–523, 2016.

[43] C. Cui, J. Shen, J. Ma, and T. Lian, "Social tag relevance estimation via ranking-oriented neighbour voting," in *Proc. of ACM Multimedia*, 2015, pp. 895–898.

[44] X. Li, C. G. Snoek, and M. Worring, "Learning social tag relevance by neighbor voting," *IEEE Trans. Multimedia*, vol. 11, no. 7, pp. 1310–1322, 2009.

[45] C. Cui, J. Ma, T. Lian, Z. Chen, and S. Wang, "Improving image annotation via ranking-oriented neighbor search and learning-based keyword propagation," *J. Am. Soc. Inf. Sci. Technol.*, vol. 66, no. 1, pp. 82–98, 2015.

[46] D. Grangier and S. Bengio, "A discriminative kernel-based approach to rank images from text queries," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 8, pp. 1371–1384, 2008.

[47] 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, pp. 64–72.

[48] Y. Pan, T. Yao, X. Tian, H. Li, and C.-W. Ngo, "Click-through-based subspace learning for image search," in *Proc. of ACM Multimedia*, 2014, pp. 233–236.

[49] 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.

[50] 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 Multimedia*, 2014, pp. 229–232.

[51] H.-T. Pu, "An analysis of failed queries for web image retrieval," *J. Inf. Sci.*, vol. 34, no. 3, pp. 275–289, 2008.

[52] L. Hollink, A. Schreiber, B. Wielinga, and M. Worring, "Classification of user image descriptions," *Int. J. Hum.-Comput. Stud.*, vol. 61, no. 5, pp. 601–626, 2004.

[53] A. Goodrum and A. Spink, "Image searching on the Excite web search engine," *Inf. Process. Manage.*, vol. 37, no. 2, pp. 295–311, 2001.

[54] Y. Liu, M. Zhang, L. Ru, and S. Ma, "Automatic query type identification based on click through information," in *Proc. of AIRS*, 2006, pp. 593–600.

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

[56] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *arXiv preprint arXiv:1207.0580*, 2012.

[57] X. Li, S. Liao, W. Lan, X. Du, and G. Yang, "Zero-shot image tagging by hierarchical semantic embedding," in *Proc. of SIGIR*, 2015, pp. 879–882.

[58] S. Cappallo, T. Mensink, and C. Snoek, "Image2emoji: Zero-shot emoji prediction for visual media," in *Proc. of ACM Multimedia*, 2015, pp. 1311–1314.

[59] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *Proc. of ICLR*, 2013.

[60] D. Metzler and W. B. Croft, "Linear feature-based models for information retrieval," *Information Retrieval*, vol. 10, no. 3, pp. 257–274, 2007.

[61] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proc. of CVPR*, 2009, pp. 248–255.

[62] Y. Lu, L. Zhang, J. Liu, and Q. Tian, "Constructing concept lexica with small semantic gaps," *IEEE Trans. Multimedia*, vol. 12, no. 4, pp. 288–299, 2010.

[63] A. Sun and S. S. Bhowmick, "Quantifying tag representativeness of visual content of social images," in *Proc. of ACM Multimedia*, 2010, pp. 471–480.

[64] G. Smith, C. Brien, and H. Ashman, "Evaluating implicit judgments from image search clickthrough data," *J. Am. Soc. Inf. Sci. Technol.*, vol. 63, no. 12, pp. 2451–2462, 2012.

[65] 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*, vol. arXiv:1408.5093, 2014.

[66] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," in *Proc. of ICLR*, 2015.

[67] 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, pp. 1–9.

[68] 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*, vol. 115, no. 3, pp. 211–252, 2015.

[69] H. Jegou, M. Douze, and C. Schmid, "Product quantization for nearest neighbor search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 1, pp. 117–128, 2011.