Image Retrieval by Cross-Media Relevance Fusion

Jianfeng Dong[1]*, Xirong Li[2,3]†, Shuai Liao[2], Jieping Xu[2,3], Duanqing Xu[1], Xiaoyong Du[2]
[1]College of Computer Science and Technology, Zhejiang University
[2]Key Lab of Data Engineering and Knowledge Engineering, Renmin University of China
[3]Multimedia Computing Lab, School of Information, Renmin University of China
danieljf24@zju.edu.cn, xirong@ruc.edu.cn

ABSTRACT

How to estimate cross-media relevance between a given query and an unlabeled image is a key question in the MSR-Bing Image Retrieval Challenge. We answer the question by proposing *cross-media relevance fusion*, a conceptually simple framework that exploits the power of individual methods for cross-media relevance estimation. Four base cross-media relevance functions are investigated, and later combined by weights optimized on the development set. With DCG_{25} of 0.5200 on the test dataset, the proposed image retrieval system secures the first place in the evaluation.

Categories and Subject Descriptors

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

Keywords

Image retrieval challenge, Cross-media relevance fusion

1. INTRODUCTION

A well-educated native speaker of English is known to have a vocabulary of about 17,000 base words [3]. Image queries as varied combinations of these words are countless. Putting subjective queries like "*happy new year*" aside, even a query with strong visual clues like "*girl with long brown hair and blue eye with eyeliner on*" remains challenging for state-of-the-art visual object recognition. By all standards, finding relevant images for an unconstrained query is difficult.

In this paper we tackle the image retrieval problem in the context of the MSR-Bing Image Retrieval Challenge (IRC). In this challenge, a contesting system is asked to produce a real-valued score on each image-query pair that reflects how relevant the query could be used to describe the given image, with higher numbers indicating higher relevance. A sample of Bing user click log with 1M images and over 11M queries, called Clickture-lite [4], is provided for developing the system. While the images come from the web, contextual information such as filename, URL, and surrounding text that could be used for relevance estimation has been removed by the task organizers. Hence, how to effectively estimate cross-media relevance for a given image-query pair is essential for conquering the challenge.

As image and query are of two distinct modalities, they have to be represented in a common space so that cross-media relevance can be computed. Depending on the choice of the common space, we categorize existing works into three groups, namely image2text, text2image, and semantic embedding. In the first group, the image is represented by a bag-of-word vector, either by label propagation from visual neighbors [12, 13] or by a deep neural network that maps images into a bag-of-words space [1]. The second group reverses the mapping direction, representing the query by a set of images. The images are retrieved either from Clickture-lite by tag-based image retrieval [2], or from the other test images of the given query [13, 14]. Consequently, the cross-media relevance is implemented by aggregating the visual similarity between the retrieved images and the given image. Lastly, semantic embedding based methods project both image and query into a *learned* space. Four methods for constructing such a space are investigated in [11], where Canonical Correlation Analysis and its variant are found to be the best.

The three groups of methods may complement each other, due to their different mechanisms for cross-media relevance estimation. By directly matching with query logs, image2text and text2image are more suited for instance search, e.g., finding images of a specific celebrity. Semantic embedding has an effect of dimension reduction and topic discovery, and thus works for category search. Following this argument, fusion of cross-media relevance scores generated by different methods seems appealing.

While efforts have been made along the line of fusion [2, 11], the fact that none of these systems has ever won the challenge is somewhat discouraging. The winning team of the IRC 2013 edition even concludes that fusion has little influence on the performance [13]. Notice that in [11], the results to be combined are all generated by semantic embedding methods. This may make the results less diverse and thus less complementary. Moreover, the fusion weights are either set to be equal [11] or chosen by hand [2]. All this

*Work performed at Renmin University of China.
†Corresponding author

MM'15, October 26–30, 2015, Brisbane, Australia.

ACM 978-1-4503-3459-4/15/10 ...$15.00.
DOI: http://dx.doi.org/10.1145/2733373.2807419.

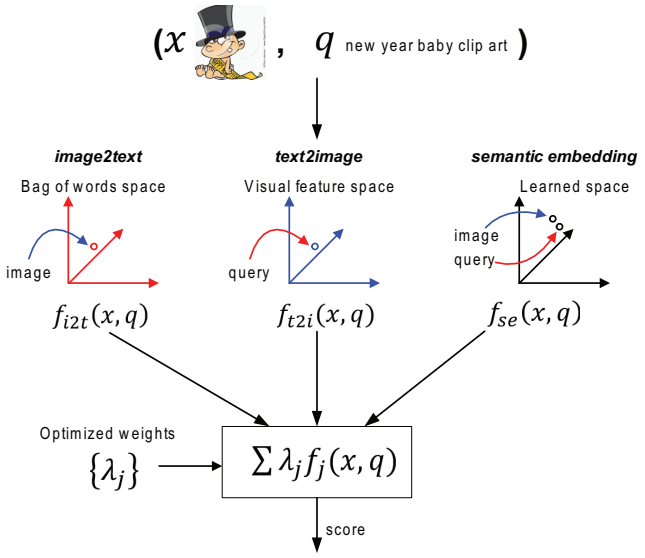

Figure 1: A conceptual diagram of the proposed cross-media relevance fusion (best viewed in color).

makes us believe that the potential of fusion has not been well explored.

Our contribution is a solution to IRC by cross-media relevance fusion, which is ranked the 1st according to the official evaluation. A conceptual diagram of the proposed solution is illustrated in Fig. 1.

The remaining sections are organized as follows. We describe the new solution in Section 2, followed by evaluation in Section 3. Conclusions are given in Section 4.

2. PROPOSED SOLUTION

Given an unlabeled image x and a textual query q, we aim to construct a real-valued function $f(x,q)$ that outputs a cross-media relevance score for the given pair. As discussed in Section 1, we opt for a fusion-based solution that exploits the power of the individual methods for cross-media relevance estimation. In what follows, we depict our choices of the methods, and present a strategy for optimizing the fusion weights.

2.1 Cross-Media Relevance Estimation

In order to generate relevance scores that complements each other, we adopt three different types of methods, i.e., image2text, text2image, and semantic embedding.

1) image2text $f_{i2t}(x,q)$. Inspired by the label propagation algorithm [12], for a given image x, we retrieve its k nearest visual neighbors, denoted as $\{x_1,\ldots,x_k\}$, from Clickture-lite. Relevance between 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. In particular, we have

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

where $sim(x,x_i)$ is a visual similarity, and

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

where m_i is the number of queries associated with x_i in Clickture-lite, $click_{i,j}$ is click count x_i received with respect to query $q_{i,j}$, and $sim(q,q')$ is a query-wise similarity. Viewing each query as a set of tags, we use the Jaccard similarity coefficient to realize $sim(q,q')$.

For the visual similarity, we extract off-the-shelf CNN features and use the cosine similarity. Specifically, we employ two pre-trained Caffe models, using their fc7 layer as features. One model was learned from examples of 1,000 visual object classes in the Large Scale Visual Recognition Challenge (LSVRC) [5], and thus describes what objects are present in an image. The second model was learned from examples of 205 scene classes [15], and thus describes what scenes the image captures. For the ease of reference, we term the two models and corresponding features as LSVRC-CNN and Places-CNN.

2) text2image $f_{t2i}(x,q)$. We propose to realize text2image in a dual form of image2text. For a given query q, we first retrieve the top k most similar queries, denoted as $\{q_1,\ldots,q_k\}$, from Clickture-lite. Relevance between x and q is computed as a weighted sum of the visual similarity between x and images associated with each of the neighbor queries. In particular, we have

$$f_{t2i}(x,q) := \frac{1}{k}\sum_{j=1}^{k} sim(q,q_j)\cdot sim_v(x,q_j), \qquad (3)$$

where

$$sim_v(x,q_j) := \frac{1}{n_j}\sum_{i=1}^{n_j} sim(x,x_{i,j})\cdot \log(click_{i,j}), \qquad (4)$$

and n_j is the number of images associated with q_j in the query log.

3) text2image as Parzen window $f_{pw}(x,q)$. As an extreme case of Eq. 3, we set the similar query to be the test query itself. In this case, the click count of the log item in Eq. 4 will not be available. Therefore, the relevance score is computed as the averaged similarity between the given image and the other test images of the given query. Such a simplified formula has played a key role in the previous winning systems [1, 13, 14]. Note that the averaged similarity is essentially a specific form of Parzen window density estimation. The Parzen window version of text2image is expressed as

$$f_{pw}(x,q) := \frac{1}{n_q}\sum_{i=1}^{n_q} K_h(x-x_i^q), \qquad (5)$$

where n_q is the number of test images corresponding to q, x_i^q indicates a specific test image, $K(\cdot)$ is the kernel parameterized by the bandwidth h. We use the normal kernel.

4) semantic embedding $f_{se}(x,q)$. We utilize ConSE [10], a deep learning based semantic embedding method originally developed for zero-shot learning. The key idea of ConSE is to first construct a semantic space by a neural language model [9] trained on millions of web documents. Each dimension of the space corresponds to a specific word, which is associated with a unique real-valued vector. Since the training process of the language model is highly scalable and efficient, the size of the vocabulary can easily be hundreds of thousands. This is an advantage compared to semantic embedding methods that have been investigated in the context of IRC, where the vocabulary size is typically limited to 50k for the scalability concern [1, 11].

An image is embedded into the same space, by first predicting relevant tags using an existing image annotation system (either LSVRC-CNN or Places-CNN in this work), and then taking the convex combination of the embedding vectors of the predicted tags. Let $\{y_1, \ldots, y_T\}$ be the top T most relevant tags predicted for a given image, the image embedding vector is obtained by

$$\mathbf{v}(x) := \frac{1}{Z} \sum_{i=1}^{T} p(y_i|x) \cdot \mathbf{v}(y_i) \quad (6)$$

where $p(y_i|x)$ is the relevance score of y_i given x, $Z = \sum_{i=1}^{T} p(y_i|x)$ is a normalization factor, and $\mathbf{v}(y_i)$ is the word embedding vector. We adopt word vectors from [6], which were trained on many Flickr tags and found to better capture visual relationships compared to word vectors learned from web documents.

To embed a query of arbitrary length, we use average pooling of the vectors of the query words. Consequently, the cross-media relevance is computed as the cosine similarity between the embedding vectors of the query and the image:

$$f_{se}(x, q) := cosine(\mathbf{v}(x), \mathbf{v}(q)). \quad (7)$$

2.2 Relevance Fusion with Optimized Weights

Given the above four methods and two CNN models, eight scores are computed per image-query pair. Without loss of generality, we use $\{f_i(x, q)|i = 1, \ldots, d\}$ to denote a d-dimensional score vector of a given pair. Linear fusion is applied for its effectiveness and efficiency:

$$f(x, q) := \sum_{i=1}^{d} \lambda_i f_i(x, q), \quad (8)$$

where $\{\lambda_i\}$ are weights to be optimized.

Formalization as Eq. 8 opens the way for a number of learning to rank algorithms. We employ Coordinate Ascent, initially developed for document retrieval [8], and later shown to be effective for combining multiple sources of ranking features for image retrieval [7]. Per iteration the algorithm optimizes a chosen weight by a bi-direction line search with increasing steps, with the remaining weights fixed. The search process requires no gradient computation, so a non-differentiable performance metric, e.g., DCG_{25} specified by IRC, can be directly optimized.

3. EVALUATION

3.1 Setup

Datasets. We use Clickture-lite as the training set to derive $f_{i2t}(x, q)$, $f_{t2i}(x, q)$, $f_{pw}(x, q)$, and $f_{se}(x, q)$, and the provided dev set for optimizing the weights $\{\lambda_i\}$. The test dataset, with 9.4K queries and 147K images, is significantly larger when compared to the previous editions. Moreover, a considerable amount of irrelevant image-query pairs are added, making the task even more challenging. As shown in Table 1, while the dev and test sets have similar upper bound, the random baseline of the test set is much lower.

Query preprocessing. We conduct standard text preprocessing: removing punctuation and lemmatizing words by NLTK. Meaningless words such as "image" and "picture" and standard English stopwords are removed also. Consequently, some queries are merged, resulting in 9M unique queries in the training set.

Table 1: Performance comparison.

Method	Dev set	Test set
random baseline	0.4702	0.4260
Upper bound	0.6852	0.6924
image2text + LSVRC-CNN	0.4992	–
image2text + Places-CNN	0.4967	–
text2image + LSVRC-CNN	0.5157	0.4897
text2image + Places-CNN	0.5086	–
Parzen window + LSVRC-CNN	0.5428	–
Parzen window + Places-CNN	0.5347	–
semantic embedding + LSVRC-CNN	0.4857	–
semantic embedding + Places-CNN	0.4795	–
fusion-feat6-avg	0.5139	–
fusion-feat6	0.5201	0.4929
fusion-feat8-avg	0.5369	–
fusion-feat8	0.5529	0.5200

Performance metric. Following the evaluation protocol, DCG_{25} is reported.

Naming convention. We name each run using the corresponding method plus feature name. E.g., image2text + LSVRC-CNN means using relevance scores produced by the image2text method with the LSVRC-CNN feature. The runs start with "fusion" correspond to cross-media relevance fusion, while fusion-feat8 means all the 8 scores are combined, and fusion-feat6 does not include two scores related to $f_{pw}(x, q)$. The difference between fusion-feat8 and fusion-feat8-avg is that the latter uses equal weights.

3.2 Results

.

Performance comparison on the dev set. As shown in Table 1, all methods outperform the random baseline. Concerning the choice of the visual features, LSVRC-CNN is better than Places-CNN. This is probably because Places-CNN focuses on scene classes, making its features less effective for representing the visual content of generic images. We attribute the relatively lower performance of semantic embedding to the fact that the LSVRC 1,000 object classes are not specifically designed for describing generic queries. The result that fusion-feat6-avg and fusion-feat8-avg are worse than the best single method shows the importance of weight optimization. Fig. 2 shows top 10 results of query "new year baby clip art" retrieved by different methods. In this example, fusion improves over Parzen Window in terms of both accuracy and diversity.

Performance comparison on the Test Set. We submitted three runs, i.e., text2image, fusion-feat6, and fusion-feat8. Our runs lead the evaluation, as shown in Fig. 3. Similar to the dev set, the Parzen window method is important. Adding it increases DCG_{25} from 0.4929 to 0.5200.

Concerning efficiency, our prototype system runs at a speed of approximately 0.8 image-query pair per second on a common GPU.

4. CONCLUSIONS

Our experiments with IRC support conclusions as follows. Among the four methods we have investigated, Parzen win-

Figure 2: Top 10 results of query "new year baby clip art" returned by different methods.

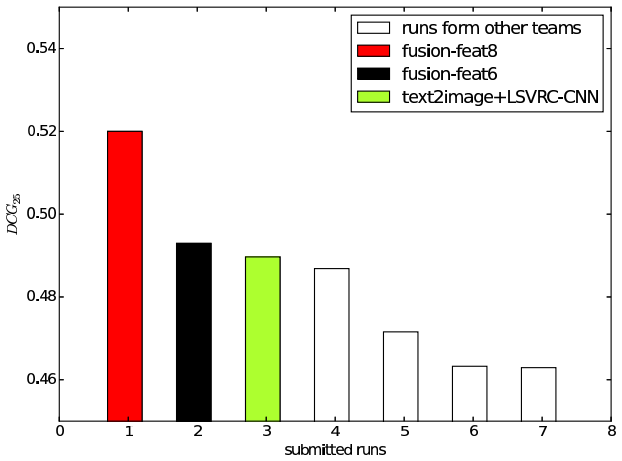

Figure 3: Performance on the test dataset. Our submissions lead the evaluation.

dow remains the most effective. Fusion leads to the best performance. But the superiority is achieved only when appropriate care is taken to optimize fusion weights.

Acknowledgments. This work was supported by the National Natural Science Foundation of China (No. 61303184), the Fundamental Research Funds for the Central Universities and the Research Funds of Renmin University of China (No. 14XNLQ01), the National Key Technology Support Program (No. 2012BAH70F02), and the National Social Science Fundation of China (No. 12&ZD141).

5. REFERENCES

[1] 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 *ACM MM*, 2014.

[2] 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.

[3] R. Goulden, P. Nation, and J. Read. How large can a receptive vocabulary be? *Applied Linguistics*, 11(4):341–363, 1990.

[4] 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. *ACM MM*, 2013.

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

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

[7] X. Li, C. Snoek, M. Worring, and A. Smeulders. Fusing concept detection and geo context for visual search. In *ICMR*, 2012.

[8] D. Metzler and B. Croft. Linear feature-based models for information retrieval. *Inf. Retr.*, 10(3):257–274, 2007.

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

[10] 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. *ICLR*, 2014.

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

[12] 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 *ACM MM*, 2013.

[13] 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 *ACM MM*, 2013.

[14] Z. Xu, Y. Yang, A. Kassim, and S. Yan. Cross-media relevance mining for evaluating text-based image search engine. In *ICME*, 2014.

[15] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. *NIPS*, 2014.