International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 11 (2015) pp. 29259-29267

http://www.ripublication.com

Novel Image Retrieval Approach in Similarity Integrated Network Using Combined Ranking Algorithm

Dr.P.Selvi[1], R.Sandya[2], T. Yogeswari [3], G.Viveka[4]

Dept of Information Technology, Tagore Engineering College, Chennai,India.
[1]p.selvirajendran@gmail.com
[2]sandya24081992@gmail.com
[3]yogeerajan@gmail.com
[4]vekagopalakrishnan@gmail.com

Abstract

In image rich websites like flickr, Photobucket retrieving images in such a large network is very useful but also very tedious process. It also consumes more time and the retrieved contents are not exactly relevant always. In this paper, three algorithms have been proposed to improve the performance of such sites. To compare the similarity of the images efficiently, HMok-SimRank algorithm is used. It's derived from similarity algorithm. Integrated Weighted Similarity Learning(IWSL) is used to integrate meta information descriptions with image content. Finally, Ranking algorithm is used to rank the images for the order of ret*rieval. Benefits of our proposed system applied in flickr are experimentally shown in terms of both relevance and speed.*

Key words – Similarity, Image retrieval, Integrated Networks

1. INTRODUCTION

In a high scale image rich networks like Picasa, Photobucket, flickr retrieving exact images based on the user query is very complex and time consuming that requires extraordinary design and implementation efforts. In text-based retrieval, estimating the similarity of the words in the context is useful for returning more relevant images. In image-content based retrieval most methods and systems compute image similarity based on image content features. Hybrid approach combines both text features and image content features together. Techniques implemented in the existing sites includes Google Distance [1], Flickr Distance [2], SimRank [3] for computing similarity, Google's VisualRank [4] for content-based retrieval and Integration algorithms [5], [6] for combining both text and image features. Existing graph-based search method also produce conflicts. The above approaches involves complexity, increased time and cost consumption, reflecting in deprived performance.The concept of image retrieval in heterogeneous image rich networks such as PhotoBucket and flickr are recommended with HMok-SimRank and IWSL. HMok-SimRank is derived from Link-based Similarity algorithm and IWSL (Integrated Weighted Similarity Learning) is for the integration of link and content similarities. This has been discussed in this paper with illustrations, analogies and the overall implementation of such a system's architecture is shown below.

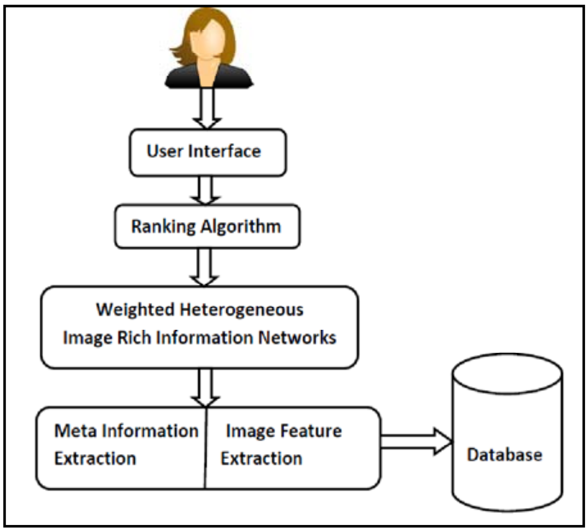

Fig. 1. Image search system architecture.

2. LINK-BASED SIMILARITY

2.1. SimRank

Node similarity is computed based on the relation between two nodes as "two nodes are similar if they are linked by similar nodes in the network". SimRank [3] is widely used for this purpose. The similarity score $S(o, o')$between two objects o and o' in a homogeneous network is as following:

$$S(o,o') = \frac{B}{|N(o)|\,N(o'|} \sum_{a \in N(o)} \sum_{b \in N(o')} S(a,b) --- (1)$$

Where $B \in [0,1]$ the damping factor, N(o) is the in-link nodes of o, $N(o')$ is the in-link nodes of o'. Here if $o = o'$, then $S(o,o')=1$ or if $N(o)=\emptyset$ or $N(o')=\emptyset$, then $S(o,o')=0$.

The storage area needed by the SimRank [3] for similarity pairs is $O(n^2)$ and the time for computation is $O(In^2P)$ for I iterations, where n is the total number of objects.

2.2. K-SimRank

To reduce the above complexities in computations we choose top 'k'(k<n) candidates of the objects in SimRank. The space complexity of k-SimRank is O(nk) and the time complexity is O(InkP) for I iterations. For such an image rich network eliciting similarity between all the images is not always necessary, hence only top k candidates are chosen. The time complexity of P in k-SimRank is $O(|N(o)||N(o')|\log(k))$, log(k) is the complexity to decide whether $N_j(o')$ is a candidate of object $N_i(o)$.

2.3. Mok-SimRank

Random top-k candidates of k-SimRank do not produce best results. Hence we minimize the k candidates to k(c) candidates. Between N(o) and $N(o')$, assign N_{big} and N_{small},compute the Mok-SimRank algorithm as given in Figure 2.

The time complexity is now reduced to P_{min}i.e., $O(InkP_{min})$ from P for 1 iterations;

$$P_{min}= \begin{cases} O\left(|N_{small}|k\log(|N_{big}|)\right): \text{if } k < |N_{big}| \\ O(|N_{small}||N_{big}|\log(k)): \text{if } k \geq |N_{big}| \end{cases} \tag{2}$$

```
for (every c ϵ Nsmall )
{
if(k<|Nbig|)  /*k denotes top-k candidates*/
for(dϵk(c))
{
if( dϵNbig)
return score; /*returns similarity score*/
else
zero; /* no similarity*/
}
else
for(dϵNbig)
{
if( dϵk(c))
return score; /*returns similarity score*/
else
zero; /* no similarity*/
}
}
```

Fig. 2. Mok-SimRank algorithm

2.4. HMok-SimRank

For weighted heterogeneous network Mok-SimRank is extended to HMok-SimRank, which contains multiple types of nodes. If the chosen candidate k(c) is totally irrelevant to the users query, then the implementation of Mok-SimRank is not useful. The images in the network are linked together into similar Groups and Tags. If 'e' is an image in the network then it is linked to $N^G(e)$ and $N^T(e)$ of Group and Tag respectively. The links of the heterogeneous network is weighted as :

1. Assign 1 as weight for all the images, giving equal importance to links initially.
2. The 'tag frequency' values is now assigned to the links based on the relevant tags resulted to the user.
3. From this weights can be given to each link in the network.

The similarity scores between images $e \epsilon V_I$ and $e' \epsilon V_I$ are defined as:

$$S_{(m+1)}(e,e') = \alpha_I S^G_{(m)}(e,e') + \beta_I S^T_m(e,e'), \tag{3}$$

With

$$S^G_m(e,e') = \frac{B^G_I}{\Omega(N^G(e)) * \Omega(N^G(e'))} \sum_{a \in N^G(e)} \sum_{b \in N^G(e')} \Psi^{ab}_{ee'} S_m(a,b), \tag{4}$$

$$S^T_m(e,e') = \frac{B^T_I}{\Omega(N^T(e)) * \Omega(N^T(e'))} \sum_{a \in N^T(e)} \sum_{b \in N^T(e')} \Psi^{ab}_{ee'} S_m(a,b) \tag{5}$$

Where $N^G(e)$ is a set of groups image e links to, $N^T(e)$ is a set of tags image e links to. α_I and β_I are the weights of link-based similarity for group and tag, respectively. We set both as 0.5 in experiment to treat them with equal importance. B^G_I and B^T_I are the damping factors. $\Omega\left(N^G(e)\right)$ is the sum of weights for the links between image e and nodes in $\left(N^G(e)\right)$

$$\Omega\left(N^G(e)\right) = \sum_{a \epsilon N^G(e)} \varpi_{ea} \tag{6}$$

$\Psi^{ab}_{ee'}$ contributes to $S(a,b)$ and for $S(e,e')$ by considering link weighting. It is defined as the multiplicative combination of the weights of the two links l_{ea} and $l_{e'b}$,

$$\Psi^{ab}_{ee'} = \varpi_{ea} * \varpi_{e'b} \tag{7}$$

Weight ϖ further can be set manually or automatically, after the initializations. Similarly, we can define and compute the link-based group and tag similarity from equations (3),(4) and (5), replacing e and e' with g and g' and t and t' for groups and tags respectively, for each pair of groups $g \epsilon V_G$ and $g' \epsilon V_G$ and tags $t \epsilon V_T$ and $t' \epsilon V_T$. The setting of parameters and the variables are similar to those in (3), (4) and (5) for groups and tags.

3. CONTENT BASED SIMILARITY

This approach checks the similarity between images based on their features. The commonly referred features are color histogram, edge histogram, colorcorrelogram, CEDD, GIST, text features, Gabor features, shape and SIFT.

3.1. Content Similarity Metric

The features of the images are checked for similarity. Represent the image as a point in D-dimension feature space, may be of single type or combination of multiple types of features.

For fixed number of dimensions integration of these types are possible. Normalize the features $F \epsilon R^D$, to a unit length as:

$$f^d = f^d_{orig} / \sum_{d=1}^{D} f^d_{orig} \tag{8}$$

Then $\mathcal{X}^2$ test statistics distance between two feature vectors F_i and F_j i.e., f^d is calculated as:

$$\chi_{ij} \equiv \chi(F_i, F_j) \equiv \sum_{d=1}^{D} c^d_{ij} \tag{9}$$

$$= \frac{1}{2} \sum_{d=1}^{D} \frac{(f_i^d - f_j^d)^2}{f_i^d + f_j^d} \tag{10}$$

When these feature vectors are normalized to a unit length, the $\mathcal{X}^2$ test distance results a value from 0 to 1. If the result is 0, then it indicates they are most similar whereas 1 indicates they are the most different.

3.2. Weighted Content Similarity Metric

In content similarity treating each dimension of the feature vector equally, performance decreases. Because, the image features may result to be similar but semantically irrelevant. So, we go for weighted content similarity which do not considers all feature dimensions as important rather it puts weight on a subset of features to give semantically similar images. To evaluate the image similarity we perform the calculation based on the $\mathcal{X}^2$ test statistic distance and a D-dimensional feature weighting vector $W=(w^1,w^2,\ldots\ldots w^D)$. We define the weighted content similarity as C^W_{ij} between images i and j as follows:

$$C^W_{ij} = 1 - \frac{1}{2} \sum_{d=1}^{D} \frac{(\omega^d f_i^d - \omega^d f_j^d)^2}{\omega^d f_i^d + \omega^d f_j^d} \tag{11}$$

$$= 1 - \sum_{d=1}^{D} \omega^{d} c_{ij}^{d} \tag{12}$$

$\mathcal{X}^2$test statistic distance is chosen to optimize the performance.

4. INTEGRATION OF LINK AND CONTENT SIMILARITIES

Implementing either link-based similarity or content-based similarity separately leads to poor results. Link-based similarity is based on human annotations and it gives unsatisfying results if the annotation is wrong, incomplete or too general. If the image is not linked to any object in information network, this method fails. The following figure shows the result based on link similarity.

Fig. 3. Images annotated by the tag "droplet", but poorly optimized.

If only content-based similarity is applied, then the results are found to be similar in features but with different semantic meaning, as illustrated below:

Fig. 4. Images with high feature similarity, but with low semantic similarity.

So we integrate the link and content based similarity approaches in our system, to achieve more optimized results.

4.1. Integration Algorithm

Integration algorithm 1 has a two stage approach. The first stage involves HMok-SimRank to compute the link based similarities of objects. The second stage is the feature learning[7],[8],[9] approach i.e., content based similarity. Now the link similarities are updated based on the new content similarity.

Integration Algorithm #1: Two-stage approach
Input: G, the image-rich integrated network.

1. Finding top k similar candidates of each objects;
2. Initialization;
3. Iterate {
4. Compute link similarity for all image pairs;
5. Compute link similarity for all group pairs;
6. Compute link similarity for all tag pairs;
7. } until converge or stop criteria satisfied;
8. Perform feature learning to update W=W_{m+1}^{*};
9. Update image similarities.

Output: S, pair-wise node similarity scores.

4.2. Integrated Weighted Similarity Learning (IWSL)

Though the above algorithm integrates link and content based similarities, it does not have a better view of the entire information network. Only the link similarity results are taken for computations of feature learning. Hence this also has the same disadvantages of using only link-based similarity without integration. To update this we move forward to Integrated Weighted Similarity Learning (IWSL). The term weight has two meanings, one for the weighted heterogeneous network and the other for weighted content features. The procedure of IWSL is described below:

Integration Algorithm #2: IWSL

Input: G, the image-rich integrated network.

1. Construct kd-tree [10] over the image features;
2. Find top k similar candidates of each object;
3. Initialization;
4. Iterate {
5. Calculate the link similarity for image pairs via HMok-SimRank;
6. Perform feature learning to update W=W_{m+1}^{*}, using either GFL or LFL [7],[8],[9];
7. (Optional) Search for new top k similar image candidates based on the new similarity weighting;
8. Update the new image similarities;
9. Compute link-based similarity for all group and tag pairs via HMok-SimRank;
10. } until converge or stop criteria satisfied.

Output: S, pair-wise node similarity scores.

5. CASE STUDY

An example from the website flickr for a query image about "ring", without using HMok-SimRank and IWSL is shown in the following pictures:

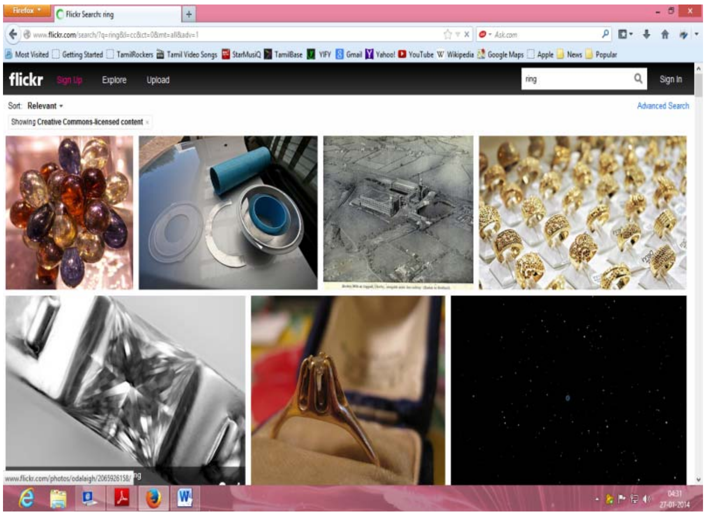

Fig. 4. flickr result for the query 'ring'.

The result of the query clearly seems to be irrelevant from the above screenshot. Our approach improvises this to give good result both semantically and visually as illustrated:

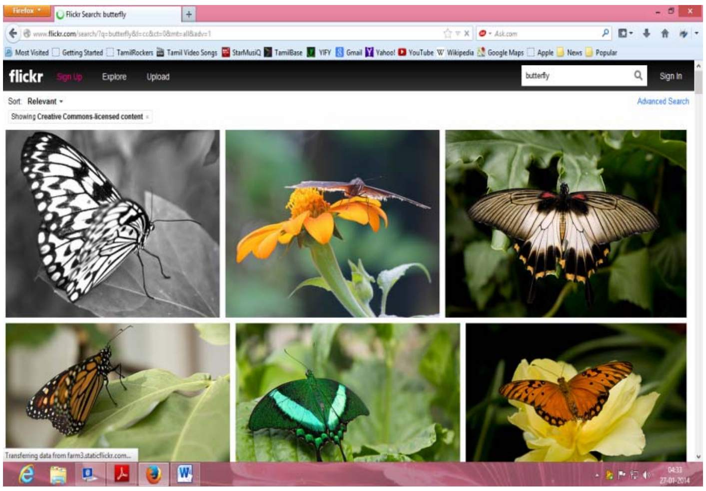

Fig. 5. flickr result for the query 'butterfly'.

6. CONCLUSION

We have presented an approach for optimizing the result in retrieving images from image-rich heterogeneous networks. We utilize HMok-SimRank for finding similar images that are linked together and weighted content similarity metric that are similar in features. To integrate these two methods we use integration algorithm IWSL (Integrated Weighted Similarity Learning). The effectiveness of our framework was demonstrated on three different stages, where our method outperformed a numberof previously used methods in complexity and computations. We have recommended a new product search system to find both visually similar and semantically relevant products.

REFERENCES

[1] R.L. Cilibrasi and P.M.B. Vitanyi, "*The Google Similarity Distanc*e," *IEEE Trans.Knowledge and Data Eng*., vol. 19, no. 3, pp. 370-383, Mar. 2007.

[2] L. Wu, X.-S.Hua, N. Yu, W.-Y. Ma, and S. Li, "Flickr Distance,"*Proc. 16th ACM Int'L conf. Multimedia*, pp. 31-40, 2008.

[3] G. Jeh and J. Widom, "SimRank: A Measure of Structural-Context Similarity," *Proc. Eighth Int'l Conf. Knowledge Discovery and Data Mining* (KDD '02), 2002.

[4] Y. Jing and S. Baluja, "VisualRank: Applying Pagerank to Large-Scale Image Search," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 30, no. 11, pp. 1877-1890, Nov. 2008.

[5] T.Deselaers and H. Mller, "Combining Textual- and Content-Based Image Retrieval, Tutorial," *Proc. 19th Int'l Conf. Pattern Recognition* (ICPR '08), http://thomas.deselaers.de/teaching/files/tutorial_icpr08/04-combinatio n.pdf, 2008.

[6] S. Sclaroff, M. La Cascia, and S. Sethi, "Unifying Textual and Visual Cues for Content-Based Image Retrieval on the World Wide Web," *Computing Vision Image Understanding*, vol. 75, pp. 86-98, July 1999.

[7] J. Tang, H. Li, G.-J. Qi, and T.-S. Chua, "Image Annotation by Graph-Based Inference with Integrated Multiple/Single Instance Representations," *IEEE Trans. Multimedia*, vol. 12, no. 2, pp. 131-141, Feb. 2010.

[8] D.G. Lowe, "Object Recognition from Local Scale-Invariant Features*," Proc. IEEE Seventh Int'l Conf. Computer Vision*, vol. 2, pp. 1150-1157, 1999.

[9] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid, "Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study," *Int'l J. Computer Vision*, vol. 73, pp. 213-238, June 2007.

[10] J.L. Bentley, "Multidimensional Divide-and-Conquer*," Comm. ACM*, vol. 23, no. 4, pp. 214-229, Apr. 1980.