

Image recommendation based on keyword relevance using absorbing Markov chain and image features

D. Sejal¹ · V. Rashmi¹ · K. R. Venugopal¹ · S. S. Iyengar² · L. M. Patnaik³

Received: 6 May 2016 / Revised: 13 June 2016 / Accepted: 16 June 2016 / Published online: 30 June 2016
© Springer-Verlag London 2016

Abstract Image recommendation is an important feature of search engine, as tremendous amount of images are available online. It is necessary to retrieve relevant images to meet the user's requirement. In this paper, we present an algorithm image recommendation with absorbing Markov chain (IRAbMC) to retrieve relevant images for a user's input query. Images are ranked by calculating keyword relevance probability between annotated keywords from log and keywords of user input query. Keyword relevance is computed using absorbing Markov chain. Images are reranked using image visual features. Experimental results show that the IRAbMC algorithm outperforms Markovian semantic indexing (MSI) method with improved relevance score of retrieved ranked images.

Keywords Annotation-based image retrieval · Content-based image retrieval · Image annotation · Image recommendation

1 Introduction

Billion of images are accessible on the Internet with expansion of the World Wide Web. The rapid growth of digital images on the Web makes it difficult for the user to find and access images of their interest. Hence, extra processing is

required to retrieve relevant images as per the user's requirement. An image retrieval system provides an effective way to retrieve a set of images to meet the users' demand.

There are two basic image retrieval techniques: (i) content-based image retrieval (CBIR) and (ii) annotation-based image retrieval (ABIR). In CBIR technique, images are retrieved based on shape, color, and texture features or using image knowledge itself rather than the meta-data associated with the image, such as descriptions, tags, or keywords. The semantic meaning of user requested query and low-level visual features of images are not matching in CBIR. Many CBIR techniques have been designed which use relevance feedback, in which search results are refined continuously by users. However, this method is impractical for very large data set, as it requires intensive computation.

In the automatic image annotation method, a computer system assigns meta-data automatically to images with keywords. Images are retrieved using this annotation. In ABIR system, semantic content is incorporated efficiently into text-based queries and image captions. Hence, many techniques are developed for automatic image annotation [1–8]. In online image retrieval identical to Google image search engine, users submit queries which consist of keywords, for searching relevant images of their interest. The search engine returns a list of images where users can click or discard the resultant images. If users are not satisfied with the retrieved images, they semantically refine the queries. Therefore, the keywords of the queries give brief, but comprehensive meaning of users need, and can be used to find relevance between annotation and user input query.

Motivation Web image search engines, such as Google and Yahoo!, retrieve images with text-based queries. These text queries are matched with textual information, such as tags, comments, surrounding text, titles, and URLs along with web

✉ D. Sejal
sej_nim@yahoo.co.in

¹ Department of Computer Science and Engineering,
University Visvesvaraya College of Engineering, Bangalore
University, Bengaluru 560001, India

² Florida International University, Miami, USA

³ National Institute of Advanced Studies, Bengaluru, India

images. Currently, only 10 % of web images have meaningful description (annotation). Although, search engine retrieves images efficiently, they are able to maintain around only 42 % precision and 12 % recall [9]. Searches do not find relevant results on Google search for 52 % of 20,000 queries [10]. This is on account of two main reasons: (i) generally, queries are short and ambiguous, i.e., the query *DM* has two different meanings *Data Mining* and *Data Mart*, and (ii) users may have different perspectives for the same query, e.g., for the query *apple*, users who are fans of apple product have different meanings than users who like the fruit apple. Therefore, it is necessary to improve image recommendations results to satisfy user's need and usability of search engine.

Contribution In this work, we have presented annotation-based image recommendation with absorbing Markov chain. Keyword relevance probability is computed for annotated keywords for all the images. Absorbing Markov chain is incorporated to find relevant link between keywords of input query with annotated keywords. Images are filtered based on their annotation similar to input query keywords. Finally, images are ranked by calculating Markov distance between user input query and annotation related to images. Furthermore, images are reranked by incorporating visual features for generated ranked image list. This method gives accurate image recommendations when per image annotation data are limited.

Organisation This paper is organised as follows: we have reviewed various content-based and annotation-based image retrieval techniques under Sect. 2. Section 3 describes Markov chain and absorbing Markov chain methods. Section 4 presents image recommendation algorithm. Section 5 discusses data collection, experiment setup, and performance evaluation. Finally, conclusions are drawn in Sect. 6.

2 Related works

In this section, we have reviewed different techniques for image retrieval. There are mainly two methods to retrieve images: content-based image retrieval (CBIR) and annotation-based image retrieval (ABIR). Images are processed with low-level visual features in CBIR system and in ABIR, images are retrieved by incorporating semantic meaning into image description and text-based queries.

2.1 Content-based image retrieval

The semantic meaning of user query and low-level visual features are not matching in the CBIR method. Hence, many CBIR techniques have been designed which uses relevance feedback, in which search results are refined continuously

by users. But users' feedbacks are very limited compared to image space's dimensionality. He et al. [11] have designed maximum margin projection (MMP) method for dimensionality reduction for image retrieval by considering top ten retrieved images as relevant images. This method maximizes negative and positive examples margin at each local surroundings. Geometrical and discriminant structures are modelled for within class and between classes in a nearest neighbour graph. Local and global visual features are derived for resulting images retrieved from Google image search [12]. Images are partitioned using kernel-based clustering to obtain multiple clusters and outliers based on visual similarity context to filter out junk images iteratively. The objective of the user query is translated using kernel learning algorithm to retrieve accurate images. This system is released for public evaluation. The method proposed in [13] does not reexamine previously checked images, as revisiting of checked images suffers from slow convergence and local maximum traps.

Web image retrieval frameworks are proposed in [14, 15] based on relevance feedback (RF). Textual and visual content-related RF is combined using a dynamic multi-modal fusion strategy and search result clustering algorithm is used to construct low-dimensional textual space for derived Web images [15]. User's click-through information is used as implicit RF for a new user, while in [14], multi-class support vector machines (SVM) are trained using image features and irrelevant images are filtered out by calculating similarity based on probabilities output from SVM and further refined based on feedback from the user.

Kekre et al. [16] have explored the mask-shape-BTC (block-truncation coding) method of image retrieval using shape features. The shape features are extracted using slope magnitude method applied on gradient of images. A method which uses Ordered-dither block-truncation coding (ODBTC) is proposed for content-based image retrieval to generate image content descriptor [17]. The ODBTC encoded data stream to construct the bit pattern and color co-occurrence image features. Both the methods require high computation to calculate features at preprocessing.

2.2 Annotation-based image retrieval

In ABIR, a computer system assigns meta-data automatically to images with keywords which can be further used to retrieve images. Hofmann et al. [18] have proposed probabilistic latent semantic analysis (PLSA) technique which intends to identify the semantic relations between the words and can be used to annotate images. This technique identifies semantic relation without recourse to a dictionary or thesaurus, text learning, information retrieval, and information filtering. PLSA has several advantages over latent semantic analysis (LSA). PLSA allows disambiguate polysems, and it calculates topical similarities by clustering words from com-

mon context. The computation time is less than LSA. Li et al. [19] have designed a hybrid model to overcome the semantic gap in image retrieval and for automatic image annotation. In this framework, continuous probabilistic latent semantic analysis (PLSA) method is used in productive stage of learning to generate visual features of images. Support vector machine classifier is used in discriminative learning stage to group multi-label data.

An ontology-based automatic image annotation framework [20] and annotation ontology-based image retrieval framework [21] have been explored. Visual properties of the images are extracted by local and global visual features [20]. Intra-concept visual dissimilarity and inter-concept visual similarity is obtained by multi-kernel learning algorithm and hierarchical boosting algorithm which uses ontology and multi-task learning. Images are annotated using concept ontology. Contextual information is extracted using annotation ontologies which are generated from different sources of information to retrieve images [21]. Task-based contextual information is also captured and reused for image management in [22].

A Wordnet and open directory project (ODP)-based image annotation system is proposed in [23,24]. Annotations are generated using neighbouring texts of the image in web page [23]. In term selection process, documents are expanded with the help of WordNet. Term selection is applied to both the document and query, so that both become more exhaustive than the original. Wang et al. [24] have presented Arista, a search-based approach for annotation that retrieves the semantics of an image by generative labels of highly related images. A hash signature and principal component analysis (PCA) model is used to generate description of an image. A novel mixture model based on open directory project (ODP) is used to understand the semantics of the image.

Image visual features are used to annotate and retrieve images [25–28]. Riad et al. [25] have presented a new majority voting technique to retrieve and annotate images with textual and visual feature. Here, each web page is considered as a voter to poll the keyword relevance to the image. Document object model (DOM) is generated for each web page based on web segmentation algorithm. Image features are extracted and images are clustered based on k -mean algorithm. Krishna et al. [26] have developed a classification-based multi-class multi-label semantic model for automatic image annotation. This model has three components: (1) feature representation, in which global features entropy, energy, and contrast features are used to describe image. (2) A classifier models is built for annotation, in which J48 decision tree is used as classifier. (3) A decision tree obtained for each concept is converted into if-then-else rules to assign the label to each image. Ayadi et al. [27] have proposed a method to annotate images automatically by computing similarity between a new medical image and old stored images.

Semantic concepts are generated by classifying image into region. Shape, texture, and color features are used to represent image regions and to create visual dictionaries [28]. This region is named with keyword and weight is allocated to it as per spatial relationship and position. Images are indexed and fetched similar to structural document retrieval.

Sang et al. [29] have designed a two-stage framework for personalized image search. It has two stages: (1) offline stage in which user-specific topic modeling is performed by ranking-based multi-correlation tensor factorization method to generate semantic topics for each user, (2) online stage in which a query is submitted by user and is mapped to user intended topics. Images are ranked by selecting users topic choice on the images which are calculated based on query distribution. Li et al. [30] have presented an algorithm by incremental model learning that simultaneously collects data set for object recognition. This method uses Bayesian incremental learning which makes use of newly added images as its theoretical base for training. This method annotates objects in the images.

Image visual information is used to improve image retrieval results in [31,32]. Pham et al. [31] have studied a method to annotate image automatically and retrieval of multimedia document using the outcome of latent semantic analysis (LSA). Documents are retrieved by combining text and image indexing. Image visual features are incorporated to improve image retrieval results. Initial retrieval results are obtained by keyword relevance model for given input query [32]. Furthermore, this retrieval result is refined with region feature vector. In the first step, semantic filtering is applied by decreasing visual information. Next, it refines by calculating similarity between visual content of query image and candidate images based on their weights.

3 Markov chain and absorbing Markov chain

3.1 Markov chain

The Markov process, a stochastic process, in which the next system state relays only on the current system state. It represents the set of states $S = \{s_1, s_2, s_3, \dots, s_n\}$, in which if the current state in s_i , then the probability of next state s_j is defined by the probability p_{ij} . These probabilities are called transition probabilities. The one-step transition probability matrix of a Markov process with N states is defined as

$$P_{N,N} = \begin{pmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,N} \\ p_{2,1} & p_{2,2} & \cdots & p_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ p_{N,1} & p_{N,2} & \cdots & p_{N,N} \end{pmatrix}$$

A Markov chain is a discrete-valued Markov process where the equilibrium vector V of a Markov chain can be

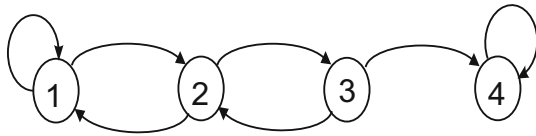


Fig. 1 Absorbing Markov chain

represented by $v \cdot P^n \approx V$, where v is any probability vector. As n approaches infinity, the Markov chain converges to a certain steady-state vector called equilibrium vector of a Markov chain. The equilibrium vector has identical rows of its states of a Markov process.

The transition probability matrix P is iteratively multiplied by itself n times to find steady-state probability vector. When the matrix size is large, then matrix multiplication is inefficient. Hence, steady-state probability vector P^n can be generated by computing eigenvalues and eigenvectors of matrix P .

The eigenvalues for a given matrix P are computed using $\det(P - \lambda I) = 0$, where λ is an eigenvalue of P , I is the identity matrix, and \det is determinant. In the case of Markov matrix, its cell values are positive and sum of every column is 1, then the largest eigenvalue is $\lambda = 1$. The eigenvector x for each λ can be solved using $(P - \lambda I)x = 0$ or $Px = \lambda x$. The power of eigenvalues is computed for any n . P^n is defined using the following:

$$P^n = \mathcal{V} \wedge \mathcal{V}^{-1}. \quad (1)$$

Here, \mathcal{V} is the matrix of eigenvectors, \wedge is the diagonal matrix of eigenvalues raised to power n , and \mathcal{V}^{-1} is the inverse matrix of \mathcal{V} .

3.2 Absorbing Markov chain

When a given state is s_i , in which $p_{ii} = 1$, then the state s_i is called absorbing state of a Markov chain. This chain is termed as absorbing Markov chain, where once we reach the state s_i , it is not possible to exit from that state, i.e., the state is absorbed [33]. Non-absorbing states are named as transient states. Absorbing Markov chain is used to compute the average time required to reach the absorbing state from any of the non-absorbing (transient) states.

Absorbing Markov chain can be represented as shown in Fig. 1. Here, states 1 and 4 are absorbing states, and 2 and 3 are transient states. The process starts from any transient state and ends up at any absorbing state in an average time.

If T_r and Ab_r represent transient and absorbing states, and t_r and a represent size of transient and absorbing states, respectively, then given transition probability matrix P is

reordered as in canonical form as $P = \begin{matrix} & T_r & Ab_r \\ T_r & Q & R \\ Ab_r & 0 & I \end{matrix}$, where I

is the $a \times a$ identity matrix, 0 is a $a \times t_r$ zero matrix, R is a non-zero $t_r \times a$ matrix representing transition probabilities from T_r to Ab_r states, and Q is a $t_r \times t_r$ matrix representing transition probabilities from T_r to T_r states.

The fundamental matrix of an absorbing Markov chain is defined as $N = (I - Q)^{-1}$ and $N = I + Q + Q^2 + \dots$. The ij th entry of N represents the expected number of times that a process reaches transient state s_j , starting from transient state s_i .

The absorption probability matrix B is the $t_r \times a$ matrix with entries b_{ij} which represent the probability that an absorbing chain is absorbed in the absorbing state s_j if it starts in transient state s_i . Then, $B = NR$, where N is the fundamental matrix and R is the canonical form.

4 Image recommendation framework and algorithm

4.1 Problem definition

Given a user input query q and annotated log lg from the web image search engine S , the objective is to recommend ranked images I_r .

4.2 Assumptions

It is assumed that the user is online while entering input query and the annotated log file is available to image search engine.

4.3 Image recommendation framework

The proposed framework presents online image retrieval similar to Google image search engine. In the retrieval phase, users submit queries which consist of keywords, to search relevant images of their interest. The search engine returns the list of images.

The framework of proposed approach is shown in Fig. 2. The framework includes two phases: (1) preprocessing phase and (2) online phase. Preprocessing phase includes step 1 and step 2. Online phase includes step 3 and step 4.

Step 1: Keyword relevance probability construction

User submits a query to the search engine and clicks on relevant images of user's interest. These query keywords give brief, but comprehensive meaning of users need. To obtain logical connection between keywords, one-step transition probability of Markov chain is computed [34]. The user clicks image IMG_i for given input query q , where keyword w_1 is followed by keyword w_2 . The current probability $p(w_1, w_2)$ is updated using Eq. 2, where each keyword represents the state of the Markov chain.

$$p(w_1, w_2) = \frac{K * p(w_1, w_2) + k}{K + k} \quad (2)$$

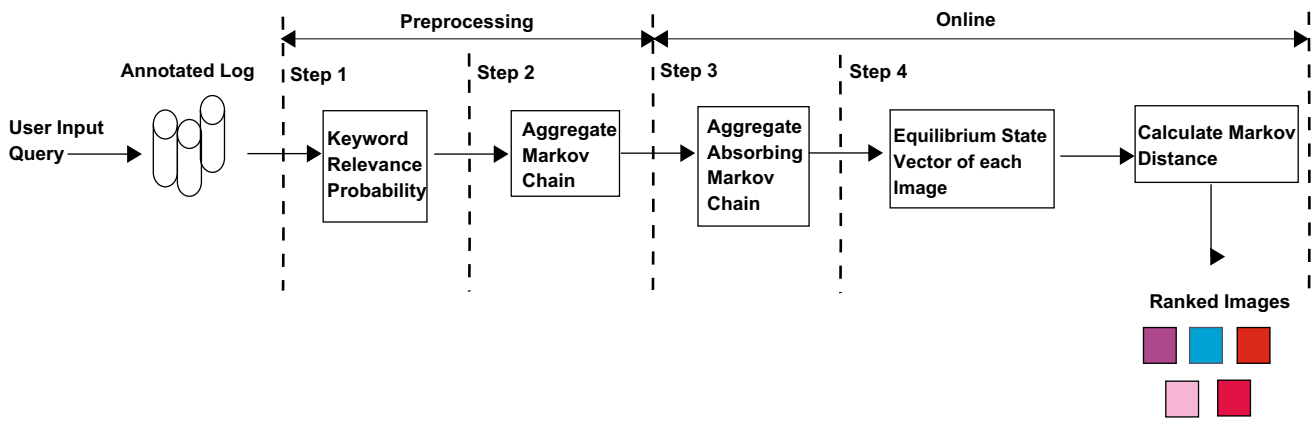


Fig. 2 Image recommendation framework

where K is the number of unique keywords and k is the number of occurrence of keyword w_1 followed by w_2 . Sometimes, images are annotated with single keyword. In such case, keyword relevance probability is calculated by considering keyword followed by itself. Therefore, relevance probability related to that image is considered as well as improved.

Whenever, a new keyword appears in the query, its initial state counter is set to 0, otherwise, it is incremented and occurrence of keyword is measured. Similarly, if that keyword is followed by another keyword, its interstate link counter is also incremented and the sequence of its occurrence is measured.

Step 2: Aggregate Markov chain construction

As users do not have enough knowledge about the information they are looking for, even in closely related images, the common keywords are very few. Hence, the state space of keywords is clustered to avoid this zero-frequency problem by constructing an aggregate Markov chain (AMC) [34]. The AMC is constructed for all the queries of all images in *log* using Eq. 2 to model keyword relevance.

The steady-state probability of AMC is computed as follows:

1. To make AMC stochastic, add small value μ to all super-diagonal elements of AMC and subtract from any non-zero elements in the same diagonal.
2. Calculate $(AMC)^n$ by calculating its eigenvalues and eigenvectors as discussed in Sect. 3.1.

Step 3: Aggregate absorbing Markov chain construction

To find relevant link between keywords of input query with annotated keywords, aggregate absorbing Markov chain (AAbMC) is incorporated. AAbMC is constructed using AMC obtained in step 2 and user input query. Keywords of input query are considered as absorbing states and other keywords in AMC represent the transient states. It is mainly focused on calculating probability by the transient states to

reach the absorbing states. Therefore, relevant link between the keywords is efficiently calculated.

Let t and r be the number of keywords in transient states(T) and absorbing states(A), i.e., keywords of input query, respectively. The canonical form of the transition matrix of AAbMC can be given as $P = \begin{matrix} T & A \\ Q & R \\ 0 & I \end{matrix}$, where Q is the $t \times t$ matrix, representing transition probabilities from keywords to keywords of transient states. R is the $t \times r$ matrix, representing transition probabilities from keywords of transient states to keywords of absorbing states. I is the $r \times r$ identity matrix. 0 is the $r \times t$ zero matrix.

The fundamental matrix for AAbMC is given by the matrix $F = (I - Q)^{-1}$. The entry f_{ij} in F represents probability of number of occurrences of keywords in transient states. The AAbMC kernel MC is the $t \times r$ matrix obtained by multiplying matrix F with R , i.e., $MC = FR$. The entry mc_{ij} in MC represents probability of transient keywords to reach absorbing keywords. It gives the required relevance between keywords of input query and other keywords of AMC.

Step 4: Markov distance calculation to rank images

In this step, the images are ranked by computing Markov distance [34] between the user input query and the annotation related to images. Images are filtered based on their annotation similar to input query keywords. The AAbMC is calculated for each filtered image and the equilibrium state vector (steady-state row vector) is obtained as discussed in Sect. 3.1.

For user input query q , the row vector i_q represents the keywords of q , and r_i represents the row vector of each image. The Markov distance d_m is calculated as given in the following:

$$d_m = (i_q - r_i)MC^T(i_q - r_i)^T. \tag{3}$$

The images are ranked based on the sorted distance values.

4.4 Algorithm

In this section, image recommendation with absorbing Markov chain (IRAbMC) algorithm is presented in Algorithm 1. It has two phases: offline and online. In offline phase, keyword relevance probability for each image is calculated from annotated log file. In the online phase, for the given user input query, keyword relevance is calculated from the offline data and keywords of input query. Finally, the images are retrieved and ranked.

Algorithm 1: IRAbMC : Image Recommendation with absorbing Markov chain

Input : Input Query q , Annotated log l

Output: Ranked Image List $I = \langle 1..k \rangle$

begin

Offline :

Compute keyword relevance probability for each image keywords in the annotated log l using equation 2.

Compute equilibrium state vector $R = \{r_1, r_2, \dots\}$ for each image as discussed in sect. III-A

Compute Aggregate Markov Chain AMC as discussed in step 2 of Sect. IV-C.

Online :

for input query q **do**

Let query keywords $A = \{w_1, w_2, \dots\}$ be the absorbing states and remaining keywords of AMC be the transition states $T = \{w_i, \text{ where } w_i \notin A\}$

Generate aggregate absorbing Markov chain $AAbMC$ using step 3 of Sect. IV-C.

Let i_q be the row vector of A .

Calculate Markov distance using i_q , R , and $AAbMC$ using equation 3.

Rank the images based on sorted distance values in ascending order.

Table 1 Statistics of data set in each cluster

Cluster name	Unique keywords	Images
Arbogreens	27	47
Australia	28	30
Barcelona	35	48
Campusinfall	36	48
Cannonbeach	34	48
Cherries	23	55
Columbiagorge	28	83
Football	76	48
Geneva	25	25
Greenlake	31	48
Indonesia	36	36
Iran	48	49
Italy	21	22
Japan	42	45
Leaflestrees	26	48
Sanjuans	49	48
Springflowers	28	48
Swissmountains	8	30
Yellowstones	39	48
Greenland	130	255

advantages over LSI and PLSI, and hence, we have compared our method with MSI.

The setup of Image recommendation with absorbing Markov chain (IRAbMC) framework and Markov semantic indexing (MSI) [34] is as follows. Aggregate Markov chain (AMC) is constructed to calculate keywords relevance probabilities between the annotated keywords. Table 3 shows the example of AMC constructed for *campusinfall* cluster from ground-truth database. A small value μ which is close to zero is added to super-diagonal elements of AMC and subtracted from any random non-zero elements within the same line [34]. In this experiment, $\mu = 0.02$ is used to make the Markov chain as monodesmic chain and retain the stochastic property of the chain. Steady-state probability of AMC, $(AMC)^n$, is calculated using eigenvalues and eigenvectors of AMC. The results are obtained by considering $n = 1$, $n = 3$, $n = 5$, $n = 7$, and $n = 10$. In IRAbMC, for user input query, aggregate absorbing Markov chain (AAbMC) is constructed by considering input query keywords as absorbing states of the $(AMC)^n$ and the remaining states as transition states. Images are filtered, such that their annotations are similar to the input query keywords, and equilibrium state vector (row vector) is calculated for those images. Markov distance is calculated between user query and row vector of filtered images. In MSI, zero mean of $(AMC)^T$ is calculated by subtracting mean row from each row of it. Covariance matrix(CM) of resulting $(AMC)^T$ is calculated. Images are

5 Experiments

5.1 Data collection

In these experiment, publicly available ground—truth database [35], in which images are already annotated, is used to evaluate the proposed method. This database has total 1109 images in 20 different clusters. Each cluster has about 55 images. Table 1 shows the statistics of data set in each cluster.

5.2 Experiment setup

Table 2 shows theoretical comparison of closely related indexing techniques for images. MSI method has several

Table 2 Comparison of indexing techniques

Method	Concept	Advantages	Disadvantages
LSI [36]	Singular value decomposition (SVD) is used to measure annotated keyword usage across images	The images are indexed by deriving concept between keyword and image statistically instead of performing lexical matching	LSI is not efficient for sparsely annotated images and requires more computation
PLSI [18]	Aspect model is used to compute conditional probability between annotated keywords and images	It identifies semantic between keywords without help of dictionary or thesaurus, text learning, information retrieval, and information filtering. It requires less computation time than LSI	Requires more annotation to improve result and aspect have over-fitting problem
MSI [34]	Aggregate Markov chain is used to calculate probability between annotated words	Calculate probability for sparsely annotated data	Images are not ranked effectively and efficiently

filtered, such that their annotations are similar to the input query keywords and equilibrium state vector (row vector) is calculated for those images. Markov distance is calculated between user query q and row vector r_i of filtered images by Eq. 4. Images are sorted and ranked based on the distance values.

$$d_m = (q - r_i)CM(q - r_i)^T. \tag{4}$$

As keyword-based search is very popular for all types of search engines, annotation-based image retrieval is very useful. Sometimes image annotations are not annotated properly, so the retrieved images are not ranked efficiently. To know the effect of visual features, we have incorporated color, texture, and shape features in retrieved ranked image of IRAbMC. For this instance, the setup is called as IRAbMC + visual features. The ranked list of images obtained in IRAbMC is considered as an informative baseline. Visual features containing RGB HSV decompose method [37], co-occurrence

matrix [37], and spatial pyramid kernel [38] are incorporated. The algorithm to rerank images with visual features obtained from IRAbMC is shown in Algorithm 2.

Algorithm 2: Rerank images with visual features

```

Input : Ranked Image List  $I = \langle 1..k \rangle$  generated from Algorithm 1, top  $q = \langle 1..m \rangle$  images from  $I$ , where  $m < k$ 
Output: Reranked images  $R = \langle 1..k \rangle$ 
begin
  for each image in  $q$  do
    Compute cosine distance  $dist$  with remaining images of  $I$  using Equation 5.
    Form a cluster with images having  $dist < \alpha$  and sort images based on the distance values in ascending order.
  Total  $C = \langle 1..m \rangle$  clusters are generated.
  Form a separate cluster  $C_s$  with images do not belongs to  $C$  and images having  $dist > \alpha$ .
  Let  $W = \{x, x \in I \text{ and } x \notin C_s\}$ .
  First, rerank images of  $q$  using smallest distance value out of  $C$ , then rerank remaining images of  $W$  and add it to  $R$ .
  Add images of  $C_s$  to  $R$  with the same order as in  $I$ .

```

The cosine distance is calculated by

$$dist = 1 - \cos(I_i, I_j) = 1 - \frac{I_i \cdot I_j}{|I_i| |I_j|} \tag{5}$$

where I_i and I_j are the normalised feature vector of images from q and I , respectively.

Reranked images are obtained by considering $q = 1$ to 5, and it is observed from results that when $q = 5$, Algorithm 2 reranks images efficiently. Hence, $q = 5$ is considered as top- q images from ranked images retrieved from IRAbMC algorithm. In this experiment, α is set to 0.01, i.e., images having cosine distance is greater than 0.01 are considered to be irrelevant and removed while clustering.

5.3 Performance evaluation

In this section, experiment results are presented and discussed. Image recommendation with absorbing Markov chain (IRAbMC) framework and Markov semantic indexing (MSI) are analyzed and compared. Experiments have been conducted on 4 GB memory and Intel(R) Core(TM) i3-3217U CPU @ 1.80GHz processor. Data set used in the experiments for IRAbMC and MSI is the same, as discussed in data collection. Top-5 image recommendations results are obtained for all 20 clusters for both the methods. A total of 100 queries, 5 from each cluster are considered for evaluation.

Table 3 AMC for *campusinfofall* cluster

	Partly	Cloudy	Sky	Building	Trees	Trunk	Bush	People	Grass	HUB	Area	Clear	Tree	Leafless	Sidewalk	Post	Quad	Ground
Partly	-	0.71	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Cloudy	-	-	0.68	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Sky	-	-	-	1.00	0.92	-	0.03	0.09	0.03	-	-	-	0.07	-	-	-	-	0.69
Building	-	-	-	-	0.98	-	-	0.03	0.76	0.07	-	-	-	0.56	0.04	-	-	-
Trees	-	-	-	0.56	0.70	0.62	0.93	0.03	0.03	-	-	-	-	0.80	0.03	-	-	-
Trunk	-	-	-	-	0.03	-	0.26	-	0.03	-	-	-	-	0.03	-	-	-	-
Bush	-	-	-	-	0.03	-	-	0.11	0.21	-	-	-	-	-	0.35	0.03	-	-
People	-	-	-	0.07	0.04	-	0.05	0.18	0.18	-	-	-	0.04	-	0.03	0.16	0.28	-
Grass	-	-	-	-	-	-	-	0.12	-	0.20	-	-	-	-	0.73	-	-	-
HUB	-	-	-	-	-	-	-	-	-	-	0.47	-	-	-	-	-	-	-
Area	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Clear	-	-	1.00	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Tree	-	-	-	-	0.99	0.04	-	0.04	0.07	-	-	-	-	-	-	-	-	-
Leafless	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Sidewalk	-	-	-	-	-	-	-	0.80	0.14	0.12	-	-	-	-	-	-	0.15	-
Post	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Quad	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Ground	-	-	-	-	-	-	-	0.47	-	-	-	-	0.04	-	-	-	-	-
Red	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Square	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Statue	-	-	-	-	-	-	0.05	-	-	-	-	-	-	-	-	-	-	-
Truck	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Water	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Ducks	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Duck	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Pond	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Fence	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Plants	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Street	-	-	-	0.09	-	-	-	-	-	-	-	-	-	-	0.03	-	-	-
Red	-	-	-	-	0.55	-	-	-	-	-	-	-	-	-	-	-	-	-
Engineering	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Car	-	-	-	-	-	-	-	0.03	-	-	-	-	-	-	-	-	-	-
Atrium	-	-	-	-	-	-	-	-	-	0.03	-	-	-	-	-	-	-	-
Sieg	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Wall	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bicycles	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Table 3 continued

	Red	Square	Statue	Truck	Water	Ducks	Duck	Pond	Fence	Plants	Street	Red	Engineering	Car	Atruim	Sieg	Wall	Bicycles
Partly	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Cloudy	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Sky	-	-	-	-	-	-	-	-	-	-	-	0.03	-	-	-	-	-	-
Building	-	-	-	-	0.11	-	-	-	-	-	0.03	0.09	-	0.03	-	-	-	0.03
Trees	0.04	-	-	-	-	-	-	-	-	-	0.09	0.08	0.03	-	-	-	0.03	-
Trunk	-	-	-	-	-	-	-	-	-	-	0.03	-	-	-	-	-	-	-
Bush	-	-	-	-	-	-	-	-	-	-	-	0.03	-	-	-	0.24	-	-
People	0.47	-	-	-	-	-	-	-	-	-	0.03	-	0.03	-	0.03	-	0.03	-
Grass	-	-	-	0.04	-	-	-	-	-	0.03	0.03	-	0.03	-	-	-	-	-
HUB	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Area	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Clear	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Tree	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Leafless	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Sidewalk	-	-	-	-	-	-	-	-	-	-	-	-	0.09	-	-	-	-	-
Post	-	-	-	-	-	-	-	-	-	-	-	-	0.03	-	-	-	-	-
Quad	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Ground	-	0.05	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Red	-	0.58	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Square	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Statue	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Truck	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Water	-	-	-	-	-	0.11	-	-	-	-	-	-	-	-	-	-	-	-
Ducks	-	-	-	-	-	-	0.04	-	0.04	-	-	-	-	-	-	-	-	-
Duck	-	-	-	-	-	-	-	-	0.10	-	-	-	-	-	-	-	-	-
Pond	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Fence	-	-	-	-	-	-	0.04	-	-	-	-	-	-	-	-	-	-	-
Plants	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Street	-	-	-	-	-	-	-	-	-	-	-	0.03	0.03	-	-	-	-	-
Red	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Engineering	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Car	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Atruim	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Sieg	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Wall	-	-	-	-	-	-	-	-	-	-	-	0.03	-	-	-	-	-	-
Bicycles	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0.03	-	-

User evaluation is performed to evaluate ranking of image recommendations for both the methods. Ten under graduate and graduate students are invited to grade the ranking results. Each student is assigned two clusters for evaluation. We have asked them to evaluate relevance between testing queries and recommended images ranking in the range of 0–1, in which 0 means totally irrelevant and 1 means totally relevant. Average values are calculated for top-1 to top-5 images. It is observed from the relevance score of ranking results that when $n = 5$,

images are ranked efficiently; hence, for comparing results, we have set $n = 5$ to calculate steady-state probability of AMC.

Figures 3, 4, 5, 6, 7 show user evaluation ranking relevance score of image recommendations results with MSI and IRAbMC methods for all the clusters. It is observed from all the graphs that for IRAbMC method images are ranked in proper order, i.e., the image relevance score is in decreasing order in all graphs for a given query. Top-1 ranked images

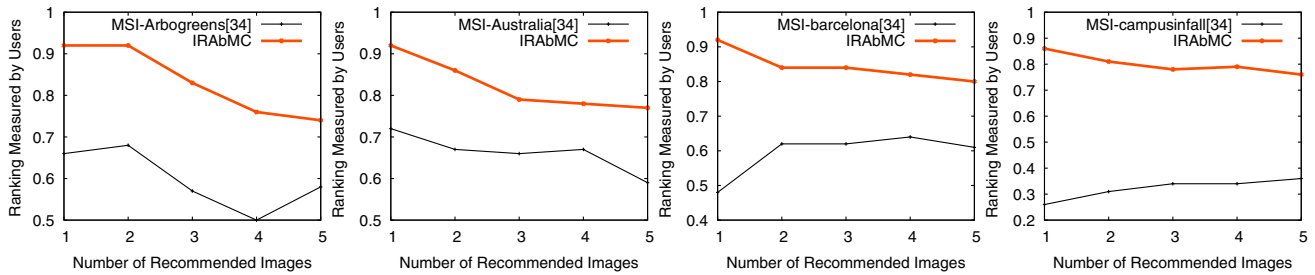


Fig. 3 User evaluation for image recommendations ranking for all the clusters. The name of cluster appears in with each graph

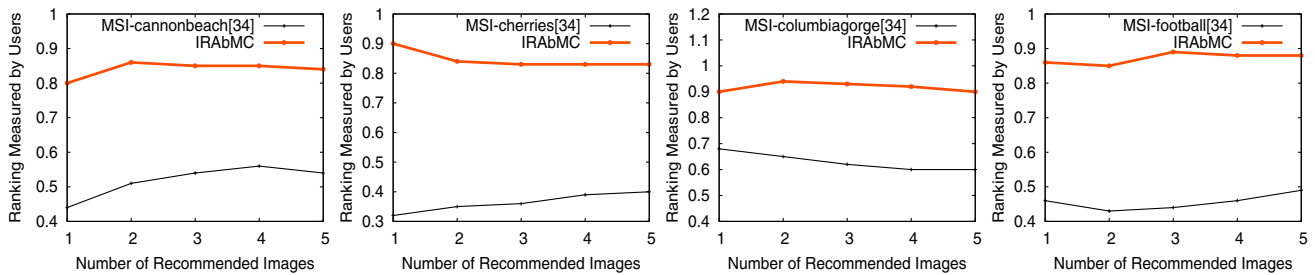


Fig. 4 User evaluation for image recommendations ranking for all the clusters. The name of cluster appears in with each graph

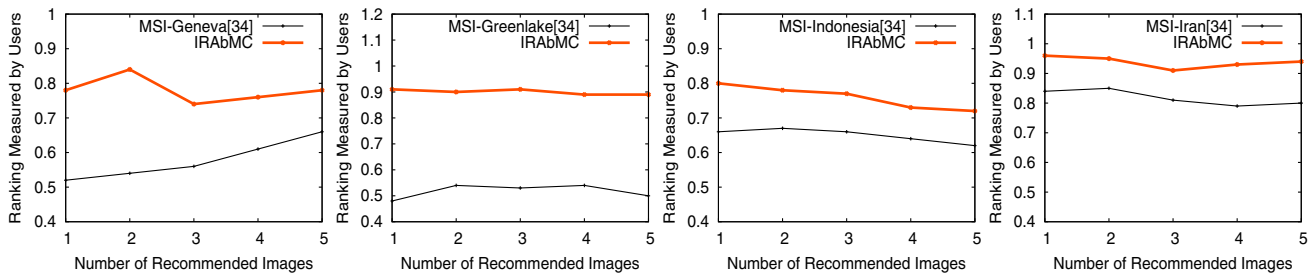


Fig. 5 User evaluation for image recommendations ranking for all the clusters. The name of cluster appears in with each graph

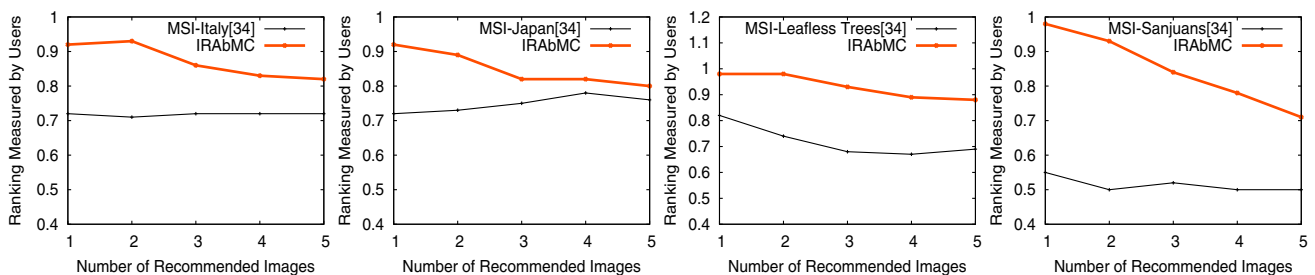


Fig. 6 User evaluation for image recommendations ranking for all the clusters. The name of cluster appears in with each graph

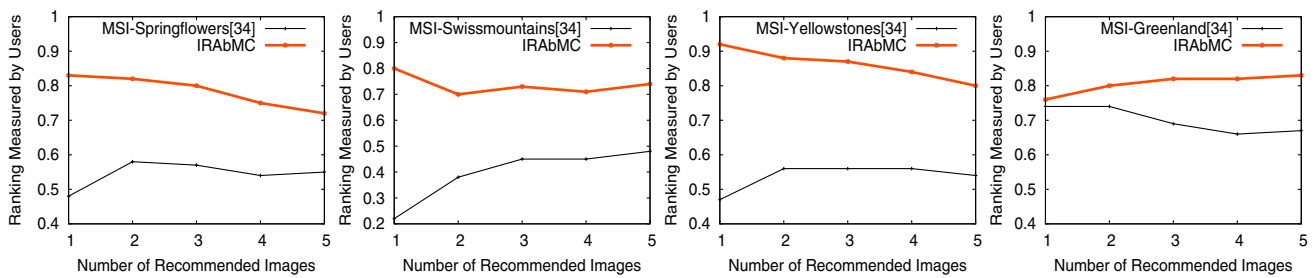


Fig. 7 User evaluation for image recommendations ranking for all the clusters. The name of cluster appears in with each graph

has highest relevance score and top-5 has least score. In MSI method, all retrieved images are relevant, but are not ranked properly. The average of overall relevance score of ranked images of all the clusters with IRAbMC method is better by 26.30 % in comparison with the MSI method.

To find out whether relevance score evaluated by students agree with each other for image recommendations, variance is computed. If variance is zero, then relevance scores are identical. Smaller variance value illustrates that relevance scores are very close to mean hence to each other. The variance of students relevance score is calculated for each testing query for both MSI and IRAbMC methods. The average variance of all testing queries for MSI method is 0.000659 and that for IRAbMC is 0.000157. It is observed from both the variance values that all the relevance scores are identical, i.e., relevance score provided for image recommendations by students agree with each other.

In Figures 8, 9, 10, 11, 12, top-5 ranked recommended images are displayed for both the methods. In all the figures, images [a–e] represent ranked images for MSI method and [f–j] represent ranked images for IRAbMC. It is observed from Fig. 8 that images [f–j] represent all the leafless trees related to given user query, but images [a–e] represent trees without leaf and also with leaf. Similarly, in Figs. 9, 10, 11, 12, images [f–j] represent relevant images with respect to

user input query, by computing proper keywords relevance of input query.

Experiments are also performed on MIRFLICKR [39] data set in which images are already annotated and 2006 images are considered to compared the recommendations results of MSI and IRAbMC methods. Figures 13 and 14 shows top-5 recommended images for queries *ocean beach* and *street art*.

Figure 15 shows the user evaluation ranking relevance score of image recommendations results with IRAbMC and IRAbMC + visual features methods for top-10 images for *australia* and *japan*. It is observed from graph that IRAbMC + visual features have a consistent ranking order compared to IRAbMC. The average of overall relevance score of ranked images of both the clusters with IRAbMC method is better by about 8 % in comparison with IRAbMC.

The IRAbMC method has quite a few advantages over MSI method which are mentioned below; IRAbMC outperforms MSI by ranking images in proper order relevant to user input query.

1. In IRAbMC, aggregate absorbing Markov chain (AAbMC) is constructed by considering input query as absorbing states. Hence, the size of the aggregate Markov chain (AMC) is reduced. Consider AMC matrix size of

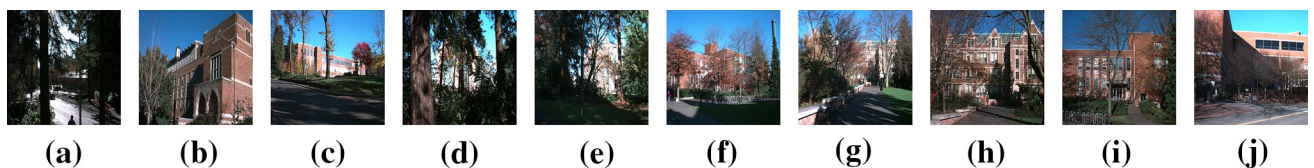


Fig. 8 Image recommendations for query *leafless trees* for cluster *campusinfall*, images a–e and f–j show result with MSI and IRAbMC method, respectively

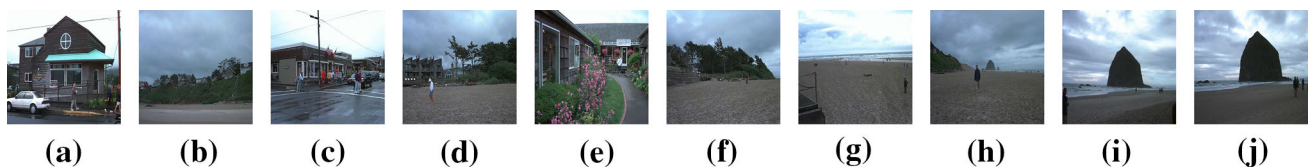


Fig. 9 Image recommendations for query *beach people* for cluster *cannonbeach*, images a–e and f–j show result with MSI and IRAbMC method, respectively

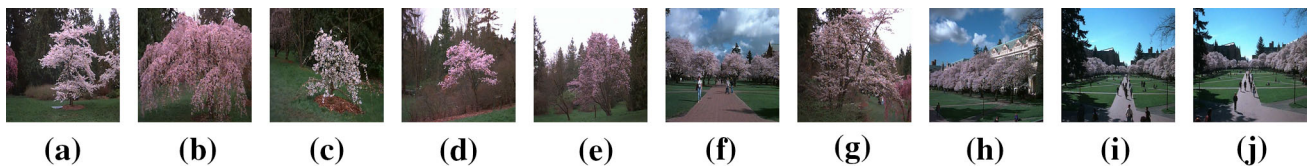


Fig. 10 Image recommendations for query *people cherry trees* for cluster *cherries*, images **a–e** and **f–j** show result with MSI and IRAbMC method, respectively

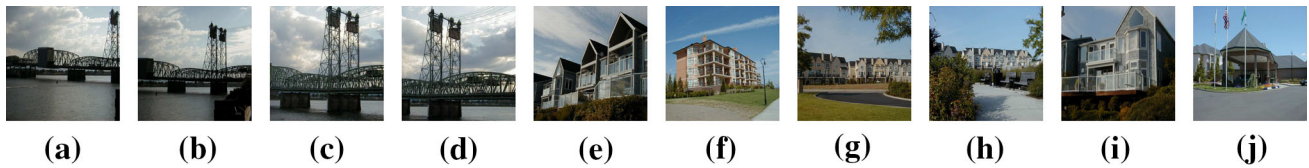


Fig. 11 Image recommendations for query *clear sky* for cluster *columbiagorge*, images **a–e** and **f–j** show result with MSI and IRAbMC method, respectively

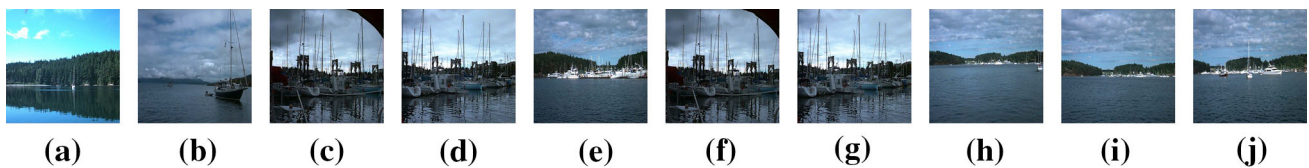


Fig. 12 Image recommendations for query *many sailboats* for cluster *sanjuans*, images **a–e** and **f–j** show result with MSI and IRAbMC method, respectively

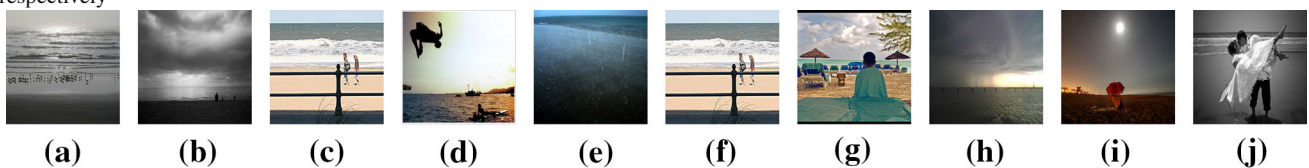


Fig. 13 Image recommendations for query *ocean beach*, images **a–e** and **f–j** show result with MSI and IRAbMC method, respectively



Fig. 14 Image recommendations for query *street art*, images **a–e** and **f–j** show result with MSI and IRAbMC method, respectively

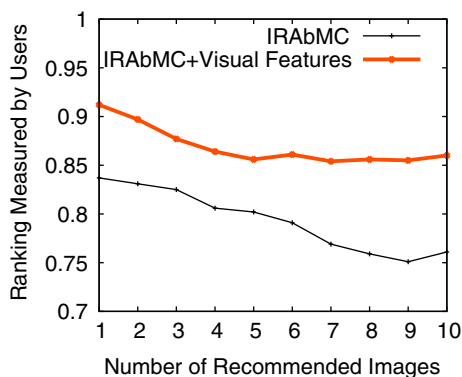


Fig. 15 User evaluation for image recommendations ranking with IRAbMC and IRAbMC + visual features method

$n \times n$ and number of absorbing states as k , then the size of resulting AAbMC is $k \times (n - k)$. As the absorbing states are the keywords of input query, k can be also less than ten, because user enters short queries.

2. Markov distance is calculated by covariance matrix in MSI and by AAbMC in IRAbMC. The covariance matrix computed from AMC remains same if the occurrence of keywords of input query is changed, while the AAbMC matrix changes. Hence, images are retrieved with the same ranking, even-though the occurrence of keywords of input query is changed in MSI method. For example, for input query *trees bush grass*, *bush trees grass*, and *grass trees bush*, the ranking of images retrieved

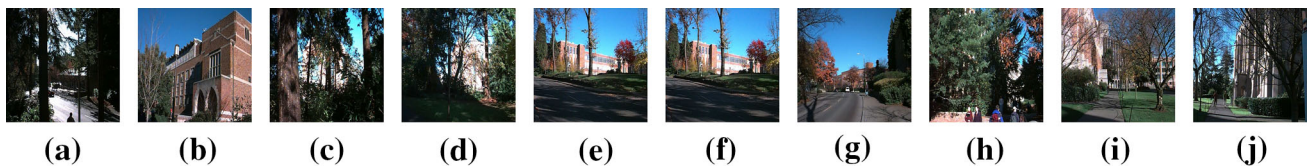


Fig. 16 Image recommendations for query *trees bush grass* for cluster *campusinfall*, images **a–e** and **f–j** show result with MSI and IRAbMC method, respectively

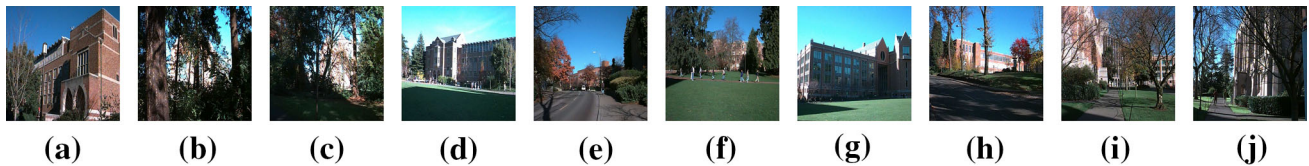


Fig. 17 Images **a–e** show ranked recommendations for query *bush trees grass*, and **f–j** show ranked recommendations for query *grass trees bush* for cluster *campusinfall* with IRAbMC method

in MSI are same, but the ranking of images differs in the IRAbMC method. In Fig. 16, images [a–e] show the ranked images for MSI method, and images [f–j] show the ranked image for IRAbMC method for input query *trees bush grass*. In Fig. 17, images [a–e] show the ranked images for input query *bush trees grass*, and images [f–j] show ranked images for input query *grass trees bush*.

3. MSI requires *AMC + Covariance Matrix + Markov Distance Calculation* and IRAbMC requires *AMC + IRAbMC + Markov Distance Calculation* to rank images. As the size of the AAbMC is reduced, the time required to recommend images is also reduced.
4. In MSI, all the images are considered to compute Markov distance, but in IRAbMC, images are filtered based on their annotation similar to input query keywords.
5. Images annotated with single keyword is not recognised while computing keyword relevance probability in MSI.

To support the reasons given above, AMC is constructed for cluster *campusinfall* as shown in Table 3, as we have displayed image recommendations results in Figs. 8, 16, and 17 of this cluster. The size of AMC matrix displayed in Table 3 is 36×36 . Markov distance is computed to filter images by covariance matrix in MSI and by AAbMC in IRAbMC. The size of covariance matrix remains same, i.e., 36×36 for all the queries for cluster *campusinfall* in MSI, while it changes in AAbMC. The AAbMC matrix is generated for input query *leafless tree* is shown in Table 4. Due to space restriction, rows and columns are interchanged. The size of the AAbMC is reduced to 2×34 .

In IRAbMC, the ranking of the images differs with occurrence of keywords of input query. Table 5 shows the AAbMC matrix for input query *grass trees bush*. Due to space restriction, rows and columns are interchanged. Similarly, for input query *trees bush grass* and *bush trees grass*, the rows are

Table 4 AAbMC for query *leafless tree* for cluster *campusinfall*

	Leafless	Trees
Partly	3.07	3.70
Cloudy	9.50	11.40
Sky	31.08	37.26
Building	−685.69	−822.04
Trunk	−735.02	−881.16
Bush	−1233.01	−1478.17
people	−469.25	−562.56
Grass	−1152.63	−1381.81
HUB	−212.83	−255.15
Area	−44.99	−53.94
Clear	14.83	17.78
Tree	−28.38	−34.03
Sidewalk	−634.86	−761.08
Post	−46.24	−55.43
Quad	−64.99	−77.91
Ground	−228.59	−274.04
Red	−154.90	−185.70
Square	−41.90	−50.23
Statue	−4.55	−5.46
Truck	−23.40	−28.05
Water	−36.50	−43.76
Ducks	−1.52	−1.82
Duck	−0.01	−0.02
Pond	−0.00	−0.00
Fence	−0.03	−0.03
Plants	−18.60	−22.30
Street	−151.96	−182.18
Red	−136.44	−163.81
Engineering	−89.34	−107.10
Car	−9.25	−11.09

Table 4 continued

	Leafless	Trees
Atruim	-6.38	-7.65
Sieg	-139.10	-166.76
Wall	-38.68	-46.37
Bicycles	-8.65	-10.37

Table 5 AAbMC for query *grass trees bush* for cluster *campusinfall*

	Grass	Trees	Bush
Partly	2.83	3.13	3.01
Cloudy	8.95	8.65	8.94
Sky	28.55	25.00	27.12
Building	-544.70	-478.57	-517.55
Trunk	-572.17	-502.69	-543.58
People	-727.69	-636.65	-689.95
HUB	-301.13	-263.58	-285.53
Area	-62.67	-54.77	-59.37
Clear	14.10	13.58	14.06
Tree	-47.81	-41.97	-45.46
Leafless	-914.91	-803.89	-869.28
Sidewalk	-1114.84	-976.30	-1057.57
Post	-70.36	-61.48	-66.65
quad	-94.70	-82.60	-89.62
Ground	-202.69	-178.10	-192.64
Red	-190.87	-166.70	-180.76
Square	-46.89	-40.88	-44.36
Statue	-4.55	-4.00	-4.33
Truck	-43.1	-37.73	-40.8
Water	-32.24	-28.31	30.62
Ducks	-1.52	-1.33	-1.44
Duck	-0.01	-0.01	-0.01
Pond	-0.00	-0.00	-0.00
Fence	-0.03	-0.02	-0.03
Plants	-34.24	-29.96	-32.46
Street	-143.71	-125.99	-136.39
Red	-124.73	-109.79	-110.62
Engineering	-117.47	-102.77	-111.35
Car	-7.89	-6.94	-7.50
Atruim	-9.36	-8.17	-8.86
Sieg	-238.57	-209.06	-226.36
Wall	-32.89	-28.83	-31.21
Bicycles	-7.68	-6.74	-7.29

interchanges based on keywords appearance in the AAbMC matrix. The AAbMC matrix for input query *trees bush grass* is similar to Table 5, but the values of row₁ shift to row₃, the values of row₂ shift to row₁, and the values of row₃ shift to row₂. Hence, the Markov distance calculated between

input query and row vector of filtered images differs with occurrence of keywords of input query. Hence, the ranking of images also differs.

6 Conclusions

In this work, we present the annotation-based image recommendation with absorbing Markov chain (IRAbMC). The Keyword relevance probability is computed for the annotated keywords for all the images. Absorbing Markov chain is incorporated to find relevant link between the keywords of input query with the annotated keywords. Images are filtered and ranked by computing Markov distance between user input query and annotation related to images. Images are reranked by incorporating visual features. Experiments are performed on publicly available data provided by the University of Washington, and results are compared with Markovian semantic indexing (MSI) method [34]. The proposed method outperforms MSI by providing more relevant images for a given user query in proper order.

References

1. Akbas E, Vural FTY (2007) Automatic image annotation by ensemble of visual descriptors. In: CVPR'07: the proceedings of IEEE conference on computer vision and pattern recognition, pp 1–8
2. Bartolini I, Ciaccia P (2010) Multi-dimensional keyword-based image annotation and search. In: The Proceedings of the 2nd international workshop on keyword search on structured data, pp 5–10
3. Wang C, Jing F, Zhang L, Zhang H-J (2007) Content-based image annotation refinement. In: CVPR'07: the proceedings of IEEE conference on computer vision and pattern recognition, pp 1–8
4. Li J, Wang JZ (2008) Real-time computerized annotation of pictures. IEEE Trans Pattern Anal Mach Intell 30(6):985–1002
5. Makadia A, Pavlovic V, Kumar S (2008) A new baseline for image annotation. Comput Vis ECCV 2008:316–329
6. Verma Y, Jawahar CV (2012) Image annotation using metric learning in semantic neighbourhoods. Comput Vis ECCV 2012:836–849
7. Wang C, Blei D, Li F-F (2009) Simultaneous image classification and annotation. In: CVPR 2009: the proceedings of IEEE conference on computer vision and pattern recognition, pp 1903–1910
8. Guillaumin M, Mensink T, Verbeek J, Schmid C (2009) Tagprop: discriminative metric learning in nearest neighbor models for image auto-annotation. In: The proceedings of IEEE 12th international conference on computer vision, pp 309–316
9. Stevenson K, Leung C (2005) Comparative evaluation of web image search engines for multimedia applications. In: ICME 2005: the proceedings of IEEE international conference on multimedia and expo, pp 4–14
10. Smyth B (2007) A community-based approach to personalizing web search. IEEE J Comput 40(8):42–50
11. He X, Cai D, Han J (2008) Learning a maximum margin subspace for image retrieval. IEEE Trans Knowl Data Eng 20(2):189–201
12. Gao Y, Peng J, Luo H, Keim DA, Fan J (2009) An interactive approach for filtering out junk images from keyword-based google search results. IEEE Trans Circuits Syst Video Technol 19(12):1851–1865

13. Liu D, Hua KA, Vu K, Yu N (2009) Fast query point movement techniques for large CBIR systems. *IEEE Trans Knowl Data Eng* 21(5):729–743
14. Rahman MdM, Antani SK, Thoma GR (2011) A learning-based similarity fusion and filtering approach for biomedical image retrieval using SVM classification and relevance feedback. *IEEE Trans Inf Technol Biomed* 15(4):640–646
15. Cheng E, Jing F, Zhang L (2009) A unified relevance feedback framework for web image retrieval. *IEEE Trans Image Process* 18(6):1350–1357
16. Kekre HB, Thepade SD, Mukherjee P, Wadhwa S, Kakaiya M, Singh S (2010) Image retrieval with shape features extracted using gradient operators and slope magnitude technique with BTC. *Int J Comput Appl* 6(8):28–33
17. Guo J-M, Prasetyo H (2015) Content based image retrieval using features extracted from halftoning-based block truncation coding. *IEEE Trans Image Process* 24(3):1010–1024
18. Hofmann T (2001) Unsupervised learning by probabilistic latent semantic analysis. *J Mach Learn* 42(2):177–196
19. Li Z, Tang Z, Zhao W, Li Z (2012) Combining generative/discriminative learning for automatic image annotation and retrieval. *Int J Intell Sci* 2(3):55–62
20. Fan J, Gao Y, Luo H (2008) Integrating concept ontology and multi-task learning to achieve more effective classifier training for multi-level image annotation. *IEEE Trans Image Process* 17(3):407–426
21. Vompras J, Scholz T, Conrad S (2008) Extracting contextual information from multiuser systems for improving annotation-based retrieval of image data. In: *The proceedings of the 1st ACM international conference on multimedia information retrieval*, pp 149–155
22. OSullivan D, Wilson DC, Bertolotto M (2011) Task-based annotation and retrieval for image information management. *Multimed Tools Appl* 54(2):473–497
23. Deniz Kılınc, Adil Alpkocak (2011) An expansion and reranking approach for annotation-based image retrieval from web. *J Expert Syst Appl* 38(10):13121–13127
24. Wang X-J, Zhang L, Ma W-Y (2012) Duplicate search based image annotation using web scale data. *IEEE J Electr Eng* 100(9):2705–2721
25. Riad A, Elminir H, Abd-Elghany S (2012) Web image retrieval search engine based on semantically shared annotation. *Int J Comput Sci Issues* 9(3):223–228
26. Krishna AN, Prasad BG (2012) Automated image annotation for semantic indexing and retrieval of medical images. *Int J Comput Appl* 55(3):26–33
27. Ayadi MG, Bouslimi R, Akaichi J (2013) A new CBIR approach for the annotation of medical images. *Int J Comput Appl* 73(6):34–45
28. Zhang D, Islam MdM, Lu G (2013) Structural image retrieval using automatic image annotation and region based inverted file. *J Visual Commun Image Represent* 24(7):1087–1098
29. Sang J, Changsheng X, Dongyuan L (2012) Learn to personalized image search from the photo sharing websites. *IEEE Trans Multimed* 14(4):963–974
30. Li L-J, Fei-Fei L (2010) Optimol: automatic online picture collection via incremental model learning. *Int J Comput Vis* 88(2):147–168
31. Pham T-T, Maillot NE, Lim J-H, Chevallet J-P (2007) Latent semantic fusion model for image retrieval and annotation. In: *The proceedings of the sixteenth ACM conference on information and knowledge management*, pp 439–444
32. Song H, Li X, Wang P (2009) Multimodal image retrieval based on annotation keywords and visual content. In: *CASE 2009: the proceedings of IEEE international conference on control, automation and systems engineering*, pp 295–298
33. Winston WL, Goldberg JB (2004) *Operations research: applications and algorithms*, vol 3. Duxbury press Belmont, CA, p 1440
34. Raftopoulos KA, Ntalianis KS, Sourlas DD, Kollias SD (2013) Mining user queries with Markov chains: application to online image retrieval. *IEEE Trans Knowl Data Eng* 25(2):433–447
35. Shapiro LG (2012) Ground truth database. Washington University. <http://imagedatabase.cs.washington.edu/groundtruth/>
36. Berry MW, Dumais ST, O'Brien GW (1995) Using linear algebra for intelligent information retrieval. *SIAM Rev* 37(4):573–595
37. Salehian H, Zamani F, Jamzad M (2012) Fast content based color image retrieval system based on texture analysis of edge map. *J Adv Mater Res* 341:168–172
38. Lux M (2011) Content based image retrieval with LIRE. In: *The proceedings of the 19th ACM international conference on multimedia*, pp 735–738
39. Huiskes MJ, Lew MS (2008) The MIR flickr retrieval evaluation. In: *MIR '08: proceedings of the 2008 ACM international conference on multimedia information retrieval*