Cross-modal Correlation Mining using Graph Algorithms

Jia-Yu Pan
Carnegie Mellon Univ.
Pittsburgh, U.S.A.

Hyungjeong Yang
Chonnam National Univ.
Gwangju, South Korea

Christos Faloutsos
Carnegie Mellon Univ.
Pittsburgh, U.S.A.

Pinar Duygulu
Bilkent Univ.
Ankara, Turkey

Abstract

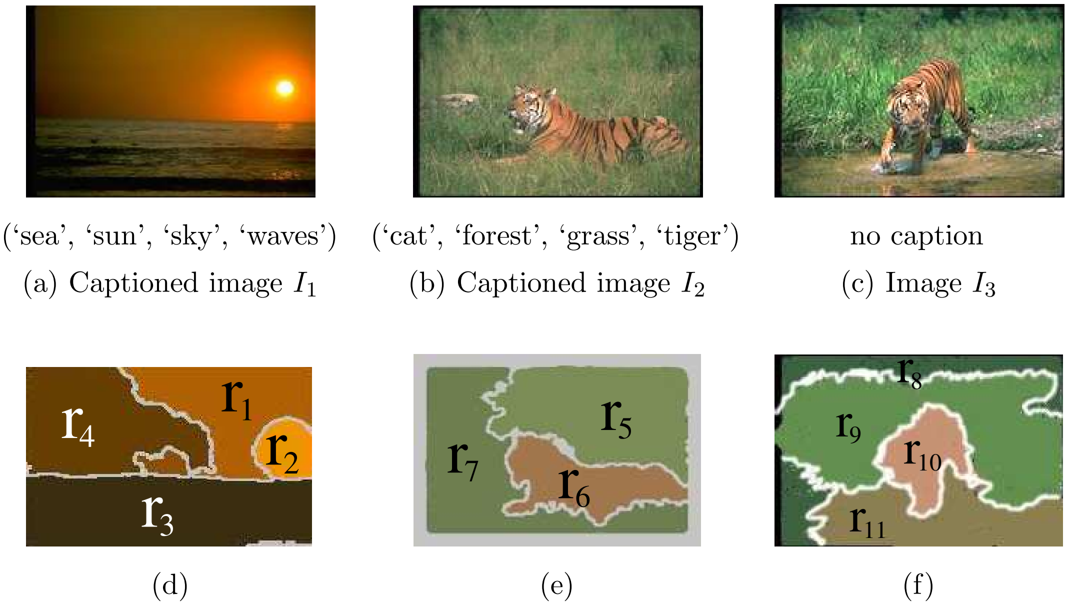
Multimedia objects like video clips or captioned images contain data of various modalities such as image, audio, and transcript text. Correlations across different modalities provide information about the multimedia content, and are useful in applications ranging from summarization to semantic captioning. For discovering cross-modal correlations, we proposed a graph-based method, *MAGIC*, which turns the multimedia problem into a graph problem, by representing multimedia data as a graph. Using "random walks with restarts" on the graph, MAGIC is capable of finding correlations among all modalities. When applied to the task of automatic image captioning, MAGIC found robust correlations between text and image and achieved a relative improvement by 58% in captioning accuracy as compared to recent machine learning techniques.

MAGIC has several desirable properties: (a) it is general and domain-independent; (b) it can spot correlations across any two modalities; (c) it is completely automatic and insensitive to parameter settings; (d) it scales up well for large datasets, (e) it enables novel multimedia applications (e.g., *group captioning*), and (f) it creates opportunity for applying graph algorithms to multimedia problems.

1 Introduction

Advances in digital technologies make possible the generation and storage of large amount of multimedia objects such as images and video clips. Multimedia content contains rich information in various modalities such as images, audios, video frames, time series, etc. However, making rich multimedia content accessible and useful is not easy. Advanced tools that find characteristic patterns and correlations among multimedia content are required for the effective usage of multimedia databases.

We called a data object which has its content presented in more than one modality a *mixed media* object. For example, a video clip is a mixed media object with image frames, audios, and other information such as transcript text. Another example is a captioned image such as a news



('sea', 'sun', 'sky', 'waves') ('cat', 'forest', 'grass', 'tiger') no caption

(a) Captioned image I_1 (b) Captioned image I_2 (c) Image I_3

(d) (e) (f)

Figure 1: Three sample images: (a),(b) are captioned with terms describing the content; (c) is an image to be captioned. (d)(e)(f) show the regions of images (a)(b)(c), respectively. Figures look best in color.

picture with an associated description, or a personal photograph annotated with a few keywords (Figure 1). In this paper, we would use the terms *medium* (plural form *media*) and *modality* interchangeably.

It is common to see correlations among attributes of different modalities on a mixed media object. For instance, a news clip usually contains human speech accompanied with images of static scenes, while a commercial has more dynamic scenes and loud background music [30]. In image archives, caption keywords are chosen such that they describe objects in the images. Similarly, in digital video libraries and entertainment industry, motion picture directors edit sound effects to match the scenes in video frames.

Cross-modal correlations provide helpful hints on exploiting information from different modalities for tasks such as segmentation [16] and indexing [7]. Also, establishing associations between low-level features and attributes that have semantic meanings may shed light on multimedia understanding. For example, in a collection of captioned images, discovering the correlations between images and caption words, could be useful for content-based image retrieval, and image annotation and understanding.

The question that we are interested in is "*Given a collection of mixed media objects, how do we find the correlations across data of various modalities?*" A desirable solution should be able to include all modalities of different properties, overcome noise in the data, and detect correlations between any subset of available modalities. Moreover, in terms of computation, we would like a method that is scalable to the database size and does not require human fine-tuning.

In particular, we want a method that can find correlations among all attributes, rather than just between specific attributes. For example, we want to find not just the image-term correlation between an image and caption terms, but also term-term and image-image correlations, using one single framework. This *any-to-any medium correlation* provides a greater picture of how attributes are correlated, e.g., "which word is usually used for images with blue top," "what words have related semantics," and "what objects appear often together in an image."

We proposed a novel, domain-independent framework, *MAGIC*, for cross-modal correlation discovery. MAGIC turns the multimedia problem into a graph problem, by providing an intuitive framework to represent data of various modalities. The proposed graph framework enables the application of graph algorithms to multimedia problems. In particular, MAGIC employs the *random walk with restarts* technique on the graph to discover cross-modal correlations.

In summary, MAGIC has the following advantages:

- It provides a graph-based framework which is domain independent and applicable to mixed media objects which have attributes of various modalities;
- It can spot any-to-any medium correlations;
- It is completely automatic (its few parameters can be automatically preset);
- It can scale up for large collections of objects.

In this study, we evaluate the proposed MAGIC method on the task of *automatic image captioning.* For automatic image captioning, the correlations between image and text are used to predict caption words for an uncaptioned image.

Application 1 (Automatic image captioning) *Given a set $\mathcal{I}_{core}$ of color images, each with caption words; and given an uncaptioned image I_{new}, find the best q (say, q=5) caption words to assign to it.*

The proposed method can also be easily extended for various related applications such as captioning images in groups, or retrieving relevant video shots and transcript words.

The paper is organized as follows. In Section 2, we discuss previous attempts on multimedia cross-modal correlation discovery. In Section 3, we introduce the proposed method MAGIC. In Section 4, we show that MAGIC achieves a better performance than recent machine learning methods on automatic image captioning (a 58% improvement on captioning accuracy). In Section 5, we discuss some system issues and show that MAGIC is insensitive to parameter settings and is robust to variability in the graph. In Section 6 , we give some conclusion remarks.

2 Related work

Multimedia knowledge representation and application have attracted much research attention recently. Mixed media objects provide opportunities for finding correlations between low-level and concept-level features [22, 4, 28], and multi-modal correlations had been shown useful for applications such as retrieval, segmentation, classification, and pattern discovery [7]. In this section, we survey previous work on cross-modal correlation modeling, as well as image captioning and news event summarization, which are our application domains that we evaluate our proposed model.

Multimedia cross-modal correlation Combining information about multimedia correlations in applications leverages all available information, and has led to improved performances in segmentation [16], classification [24, 8], retrieval [43, 46, 41], and topic detection [44, 11]. One crucial step of fusing multi-modal correlations into applications is to detect, extract and model the correlations from data.

Previous approaches on multimedia correlation modeling employ various techniques such as linear model [22, 39], graphical model [4, 28], statistical model [44, 16, 8], meta-classifier [43, 24], graph partitioning [45, 11], and link analysis [41]. While some of these works proposed general models multimedia correlation modeling, and evaluated the quality of the models by applying to some application domains [22, 4, 28], most of these works designed specific approaches for particular applications (e.g., [44, 43, 45]) which allow them to combine multi-modal information, as well as leverage domain knowledge to boost performance. In this paper, we introduce a general multimedia correlation framework, which is applied to image captioning and video summarization.

Previous work on cross-modal correlation modeling attempts to discover correlations between low-level features [22] or mid-level concepts [4, 28]. In [22], a linear model is designed to represent correlations between raw features of different modalities. In contrast, in [4, 28], the multi-modal information is first classified into concepts (e.g., “human” or “explosion”), and then the interaction between concepts are modeled using graphical frameworks [4, 28]. These approaches assume that certain domain knowledge is given, including the specification of the concepts of interest, the basic relations among concepts, etc.

To achieve the best performance, most of the previous studies require domain knowledge on select appropriate parameter values, and may involve a training phase on a labeled training set. For example, one may need to specify in details of concepts (what is a “person”), or fine-tune the mid-level features (which clustering algorithm to use). For classifier-based approaches, decisions about the type and parameters (kernel, etc.) of the classifier have to be made.

Ideas in link analysis have been explored for modeling cross-modal correlations. In [41], a similarity function for web images and another one for text blocks are defined initially. The relation between web images and the surrounding text blocks (cross-modal links) are then utilized to adjust the similarity distance between images for better performance on image retrieval. The initial similarity functions between objects (web images, text blocks), which may be complicated and difficult to obtain.

Statistical modeling of cross-modal correlation usually requires a training phase on a labeled training set. Preparing a training set where cross-modal associations are fully labeled is not easy. Moreover, the statistical model may be complex (with many parameters to be trained) and be computationally costly to train.

Our proposed framework, MAGIC, does not need a training phase, and has fewer parameters to tune. In fact, as we show later, the results are insensitive to parameter values in our experiments (Section 5.1). MAGIC uses a graph to represent the relations between objects and low-level attribute values. By relating multimedia objects via the constituent single-modal domain tokens, MAGIC avoid detailed specifications of concepts or complicate similarity functions.

Image captioning Although a picture is worth a thousand words, extracting the abundant information from an image is not an easy task. Computational techniques are able to derive low-to-mid level features (e.g., texture and shape) from pixel information, however, the gap still exists between mid-level features to concepts used by human reasoning [36, 47, 46]. One consequence of this semantic gap in image retrieval is that the user's need is not properly matched by the retrieved images, and may be part of the reason that practical image retrieval is yet to be popular.

Automatic image captioning, where the goal is to predict caption words to describe image content, is one research direction to bridge the gap between concepts and low-level features. Previous work on image captioning employs various approaches such as linear models [33, 27], classifiers [26], language models [40, 10, 17], graphical models [3, 5], statistical models [23, 18, 13], and a framework with user involvement [42].

Most previous approaches derive features from image regions (regular grids or blobs [10]), and construct a model between images and words based on a reference captioned image set. Images in the reference set are captioned by human experts, however, there is no information of the associations between individual regions and words. Some approaches attempt to explicitly infer the correlations between regions and words [10], with enhancements that take into consideration interactions between neighboring regions in an image [23]. Alternatively, there are methods which model the collective correlations between regions and words of an image [34, 35].

Comparing the performance of different approaches is not easy. Several benchmark data sets are available, however, not all previous work reports results on the same subset of images. On the other hand, various metrics such as accuracy, term precision and recall, and mean average precision have been used to measure the performance. Since the perception of an image is subjective, some work also reports user evaluation of the captioning result.

In Section 4, our proposed method, MAGIC, is applied to automatic image captioning. The correlations between words and images are detected and applied to predict caption words of a previously unseen image. To better evaluate our approach, we conduct experiments on the same data sets and report using the same performance metric which are also used in other previous works [40, 10, 3, 5].

3 Proposed method: graph-based correlation detection model

Our proposed method for mixed media correlation discovery, MAGIC, provides a graph-based representation for data attributes of various modalities, and a technique for finding any-to-any medium correlation, which is based on random walks on the graph. In this section, we explain how to generate the graph representation and how to detect cross-modal correlations.

3.1 MAGIC graph (G_{MAGIC})

In relational database management systems, a multimedia object is usually represented as a vector of m features/attributes [12]. The attributes must be *atomic* (i.e., taking single values) like "size" or "the amount of red color" of an image. However, for mixed media data sets, the attributes can be *set-valued*, such as the caption of an image (a set of words) or the image regions.

Finding correlations among set-valued attributes is not easy: Elements in a set-valued attribute could be noisy or missing altogether: regions in an image are not perfectly identified (noisy regions); the image caption may be incomplete, leaving out some aspects of the content. Set-valued attributes of an object may have different numbers of elements, and there is no given alignment between set elements. For instance, an image may have unequal numbers of caption words and regions, where a word may describe multiple regions and a region may be described by zero or more than one word.

We assume that the elements of a set-valued attribute are tokens drawn from a *domain*. We propose to gear our method toward set-valued attributes, because they include atomic attributes as a special case; and they also smoothly handle the case of missing values (null set).

Definition 1 (Domain and domain token) *The* **domain** D_i *of (set-valued) attribute* i *is a collection of atomic values, which we called* **domain tokens**, *which are the values that attribute* i *can take.*

A domain can consist of categorical values, numerical values, or numerical vectors. For example, for automatic image captioning, we have objects with m=2 attributes. The first attribute, "caption", has a set of categorical values (English terms) as its domain ; the second attribute, "regions", is a set of image regions, each of which is represented by a p-dimensional vector of p features derived from the region (e.g., color histogram with p colors). As described later in Section 4, we extract p=30 features from each region. To establish the relation between domain tokens, we assume that we have a similarity function for each domain. Domain tokens are usually simpler than mixed media objects, and therefore, it is easier to define similarity functions on domain tokens than on mixed media objects.

Assumption 1 *For each domain D_i $(i = 1, \ldots, m)$, we are given a similarity function $Sim_i(*, *)$ which assigns a score to a pair of domain tokens.*

For example, for the attribute "caption", the similarity function could be 1 if the two tokens are identical, and 0 if they are not.

Perhaps surprisingly, with Definition 1 and Assumption 1, we can encompass all the applications mentioned in Section 1. The main idea is to represent all objects and their attributes (domain tokens) as nodes of a *graph*. For multimedia objects with m attributes, we obtain a $(m + 1)$-layer graph. There are m types of nodes (one for each attribute), and one more type of nodes for the objects. We call this graph a MAGIC graph (G_{MAGIC}). We put an edge between every object-node and its corresponding attribute-value nodes. We call these edges *object-attribute-value links* (OAV-links).

Furthermore, we consider that two objects are similar if they have similar attribute values. For example, two images are similar if they contain similar regions. To incorporate such information into the graph, our approach is to add edges to connect pairs of domain tokens (attribute values) that are similar, according to the given similarity function (Assumption 1). We call edges that connect nodes of similar domain tokens *nearest-neighbor links* (NN-links).

We need to decide on a threshold for "closeness" when adding NN-links. There are many ways to do this, but we decide to make the threshold adaptive: each domain token is connected to its k nearest neighbors. We discuss the choice of k in Section 5.1, as well as the sensitivity of our results to k. Computing nearest neighbors can be done efficiently, because we already have the similarity function $Sim_i(*, *)$ for any domain D_i (Assumption 1).

We illustrate the construction of G_{MAGIC} graph by the following example.

Example 1 *For the images $\{I_1, I_2, I_3\}$ in Figure 1, the MAGIC graph (G_{MAGIC}) corresponding to these images is shown in Figure 2. The graph has three types of nodes: one for the image objects I_j's $(j = 1, 2, 3)$; one for the regions r_j's $(j = 1, \ldots, 11)$, and one for the terms $\{t_1, \ldots, t_8\}$={sea, sun, sky, waves, cat, forest, grass, tiger}. Solid arcs are the object-attribute-value links (OAV-links), and dashed arcs are the nearest-neighbor links (NN-links).*

In Example 1, we consider only k=1 nearest neighbor, to avoid cluttering the diagram. Because the nearest neighbor relationship is not symmetric and because we treat the NN-links as un-directional, some nodes are attached to more than one link. For example, node r_1 has two NN-links attached: r_2's nearest neighbor is r_1, but r_1's nearest neighbor is r_6. There is no NN-link

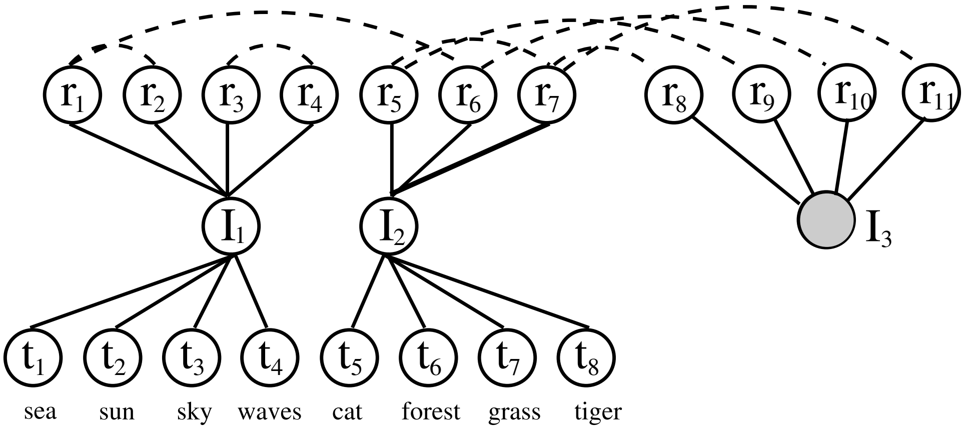


Figure 2: MAGIC graph (G_{MAGIC}) corresponds to the 3 images in Figure 1. Solid edges: OAV-links; dash edges: NN-links.

Input:

1. $\mathcal{O}$: a set of n objects (objects are numbered from 1 to n).

2. $\mathcal{D}_1$, …, $\mathcal{D}_m$: the domains of the m attributes of the objects in $\mathcal{O}$.

3. $Sim_1(*,*)$, …, $Sim_m(*,*)$: the similarity functions of domains $\mathcal{D}_1$, …, $\mathcal{D}_m$, respectively.

4. k: the number of neighbors a domain token connects to.

Output:

G_{MAGIC}: a MAGIC graph with a $(m+1)$-layer structure.

Steps:

1. Create n nodes (the object nodes), one for each object. These nodes form the layer 1.

2. For each domain $\mathcal{D}_i$, for $i = 1, \ldots, m$.

   (2.1) Let n_i be the number of tokens in the domain $\mathcal{D}_i$.

   (2.2) Create n_i nodes (the token nodes), one for each domain tokens in $\mathcal{D}_i$. This is the $(i+1)$-th layer.

   (2.3) Construct the OAV-links from the object nodes to these token nodes.

   (2.4) Construct the NN-links between the token nodes.

3. Output the final $(m+1)$-layer graph, with $N = n + \sum_{i=1}^{m} n_i$ nodes, and the OAV-links and NN-links.

Figure 3: Algorithm: G_{MAGIC}= buildgraph($\mathcal{O}$, $\{\mathcal{D}_1$, …, $\mathcal{D}_m\}$, $\{Sim_1(*,*)$, …, $Sim_m(*,*)\}$, k)

between term-nodes, due to the definition of its similarity function: 1, if the two terms are the same; or 0 otherwise. Figure 3 shows the algorithm for constructing a MAGIC graph.

We use image captioning only as an illustration: the same framework can be generally used for other problems. To solve the automatic image captioning problem, we also need to develop a

Symbol	Description
Sizes	
n	The number of objects in a mixed media data set.
m	The number of attributes (domains).
N	The number of nodes in G_{MAGIC}.
E	The number of edges in G_{MAGIC}.
k	Domain neighborhood size: the number of nearest neighbors that a domain token is connected to.
c	The restart probability of RWR (random walk with restarts, RWR).
$\mathcal{D}_i$	The domain of the i-th attribute.
$Sim_i(*, *)$	The similarity function of the i-th domain.
Image captioning	
$\mathcal{I}_{core}$	The given captioned image set (the core image set).
$\mathcal{I}_{test}$	The set of to-be-captioned (test) images.
I_{new}	An image in $\mathcal{I}_{test}$.
G_{core}	The subgraph of G_{MAGIC} containing all images in $\mathcal{I}_{core}$ (Section 4).
G_{aug}	The augmentation to G_{core} containing information of an image I_{new} (Section 4).
$\mathcal{GW}$	The gateway nodes, nodes in G_{core} that adjacent to G_{aug} (Section 4).
Random walk with restarts (RWR)	
$\mathbf{A}$	The (column-normalized) adjacency matrix.
$\vec{\mathbf{v}}_{\mathcal{R}}$	The restart vector of the set of query objects $\mathcal{R}$, where components correspond to query objects have value $1/\|\mathcal{R}\|$, while others have value 0).
$\vec{\mathbf{u}}_{\mathcal{R}}$	The RWR scores of all nodes with respect to the set of query objects $\mathcal{R}$.
$\vec{\mathbf{v}}_q$, $\vec{\mathbf{u}}_q$	$\vec{\mathbf{v}}_{\mathcal{R}}$ and $\vec{\mathbf{u}}_{\mathcal{R}}$ for the singleton query set $\mathcal{R}=\{q\}$.
$\vec{\mathbf{v}}_{\mathcal{GW}}$, $\vec{\mathbf{u}}_{\mathcal{GW}}$	$\vec{\mathbf{v}}_{\mathcal{R}}$ and $\vec{\mathbf{u}}_{\mathcal{R}}$ for RWR restarting from the gateway nodes $\mathcal{GW}$.

Table 1: Summary of symbols used in the paper

method to find good caption words - words that correlate with an image, using the G_{MAGIC} graph. This means that, for example, for image I_3, we need to estimate the affinity of each term (nodes t_1, $\ldots$, t_8) to node I_3. The terms with the highest affinity to image I_3 will be predicted as its caption words.

Table 1 summarizes the symbols we used in the paper.

3.2 Correlation detection with random walks

Our main contribution is to turn the cross-modal correlation discovery problem into a graph problem. The previous section describes the first step of our proposed method: representing set-valued mixed media objects in a graph G_{MAGIC}. Given such a graph with mixed media information, *how do we detect the cross-modal correlations in the graph?*

We define that a node A of G_{MAGIC} is correlated to another node B if A has an "affinity" for B. There are many approaches for ranking all nodes in a graph by their "affinity" for a reference node. We can tap the sizable literature of graph algorithms and use off-the-shelf methods for assigning importance to vertices in a graph. These include the electricity based approaches [29, 9], random walks (PageRank, topic-sensitive PageRank) [6, 15], hubs and authorities [21], elastic springs [25] and so on. Among them, we propose to use *random walk with restarts* (RWR) for estimating the affinity of node B with respect to node A. However, the specific choice of method is orthogonal to our framework.

The "random walk with restarts" operates as follows: To compute the affinity $u_A(B)$ of node B for node A, consider a random walker that starts from node A. The random walker chooses randomly among the available edges every time, except that, before he makes a choice, he goes back to node A (restart) with probability c. Let $u_A(B)$ denote the steady state probability that our random walker will find himself at node B. Then, $u_A(B)$ is what we want, the affinity of B with respect to A. We also call $u_A(B)$ the *RWR score* of B with respect to A. The algorithm of computing RWR scores of all nodes with respect to a subset of nodes $\mathcal{R}$ is given in Figure 4.

Definition 2 (RWR score) *The RWR score, $u_A(B)$, of node B with respect to node A is the steady state probability of node B, when we do the random walk with restarts from A, as defined above.*

Let $\mathbf{A}$ be the adjacency matrix of the given graph G_{MAGIC}, where columns of the matrix are normalized such that each sums up to 1. Let $\vec{\mathbf{u}}_q$ be a vector of RWR scores of all N nodes, with respect to a restart node q. Let $\vec{\mathbf{v}}_q$ be the "restart vector", which has all N elements zero, except for the entry that corresponds to node q, which is set to 1. We can now formalize the definition of RWR scores (Definition 3).

Definition 3 (RWR scores computation) *The N-by-1 steady state probability vector $\vec{\mathbf{u}}_q$, which contains the RWR scores of all nodes with respect to node q, satisfies the equation:*

$$\vec{\mathbf{u}}_q = (1-c)\mathbf{A}\vec{\mathbf{u}}_q + c\vec{\mathbf{v}}_q, \tag{1}$$

where c is the restart probability of the RWR from node q.

The computation of RWR scores can be done efficiently by matrix multiplication (Step 4.1 in Figure 4), with computational cost scales linearly with the number of elements in the matrix $\mathbf{A}$, i.e., the number of graph edges determined by the given database. In general, the computation of RWR scores converges after a few (~ 10) iterations (Step 4 in Figure 4). In our experiments, each RWR computation takes less than 5 seconds. Therefore, the computation of RWR scales well with the database size. Fortunately, MAGIC is modular and can continue improve its performance by including the best module [19, 20] for fast RWR computation.

```
Input:
1. $G_{MAGIC}$: a MAGIC graph with $N$ nodes (nodes are numbered from 1 to $N$).
2. $\mathcal{R}$: a set of restart nodes. (Let $|\mathcal{R}|$ be the size of $\mathcal{R}$.)
3. $c$: the restart probability.
Output:
$\vec{\mathbf{u}}_{\mathcal{R}}$: the RWR scores of all nodes with respect to $\mathcal{R}$
Steps:
1. Let $\mathbf{A}$ be the adjacency matrix of $G_{MAGIC}$. Normalize the columns of $\mathbf{A}$ and make each column sum up to 1.
2. $\vec{\mathbf{v}}_{\mathcal{R}}$ is the $N$-by-1 restart vector, whose $i$-th element $\vec{\mathbf{v}}_{\mathcal{R}}(i)$ is $\frac{1}{|\mathcal{R}|}$, if node $i$ is in $\mathcal{R}$; otherwise, $\vec{\mathbf{v}}_{\mathcal{R}}(i)$=0.
3. Initialize $\vec{\mathbf{u}}_{\mathcal{R}}$=$\vec{\mathbf{v}}_{\mathcal{R}}$.
4. while($\vec{\mathbf{u}}_{\mathcal{R}}$ has not converged)
  4.1 Update $\vec{\mathbf{u}}_{\mathcal{R}}$ by $\vec{\mathbf{u}}_{\mathcal{R}}$ = (1-c)$\mathbf{A}\vec{\mathbf{u}}_{\mathcal{R}}$ + c$\vec{\mathbf{v}}_{\mathcal{R}}$
5. Return the converged $\vec{\mathbf{u}}_{\mathcal{R}}$.
```

Figure 4: Algorithm: $\vec{u}_{\mathcal{R}}$ = RWR(G_{MAGIC}, $\mathcal{R}$, c)

The RWR scores specify the correlations across different media and could be useful in many multimedia applications. For example, to solve the image captioning problem for image I_3 in Figure 1, we can compute the RWR scores $\vec{\mathbf{u}}_{I_3}$ of all nodes and report the top few (say, 5) term-nodes as caption words for image I_3. Effectively, MAGIC exploits the correlations across images, regions and terms to caption a new image.

The RWR scores also enable MAGIC to detect any-to-any medium correlation. In our running example of image captioning, an image is captioned with the term nodes of highest RWR scores. In addition, since all nodes have their RWR scores, other nodes, say image nodes, can also be ranked and sorted, for finding images that are most related to image I_3. Similarly, we can find the most relevant regions. In short, we can restart from any subset of nodes, say term nodes, and derive term-to-term, term-to-image, or term-to-*any* correlations. We discuss more on this in Section 4.3. Figure 5 shows the overall procedure of using MAGIC for correlation detection.

4 Application: Automatic image captioning

Cross-modal correlations are useful for many multimedia applications. In this section, we present results of applying the proposed MAGIC method to automatic image captioning [34, 35]. Intuitively, the cross-modal correlations discovered by MAGIC are used in the way that an image is captioned automatically with words that correlated with the image content.

```
Step 1: Identify the objects $\mathcal{O}$ and the $m$ attribute domains $\mathcal{D}_i$, $i = 1, \ldots, m$.
Step 2: Identify the similarity functions $Sim_i(*, *)$ of each domain.
Step 3: Determine $k$: the neighborhood size of the domain tokens. (Default value $k = 3$.)
Step 4: Build the MAGIC graph,
    $G_{MAGIC}$= buildgraph($\mathcal{O}$, $\{\mathcal{D}_1, \ldots, \mathcal{D}_m\}$, $\{Sim_1(*, *), \ldots, Sim_m(*, *)\}$, $k$).
Step 5: Given a query node $\mathcal{R}$=$\{q\}$ ($q$ could be an object or a token),
    (Step 5.1) Determine the restart probability $c$. (Default value $c = 0.65$.)
    (Step 5.2) compute the RWR scores:
        $\vec{u}_{\mathcal{R}}$ = RWR($G_{MAGIC}$, $\mathcal{R}$, $c$).
Step 6: Objects and attribute tokens with high RWR scores are correlated with $q$.
```

Figure 5: Instructions for detecting correlations using MAGIC. Functions "buildgraph()" and "RWR()" are given in Figures 3 and 4, respectively.

We evaluate the quality of the cross-modal correlations by MAGIC in terms of the captioning accuracy. We show experimental results to address the following questions:

- Quality: Does MAGIC predict the correct caption terms?
- Generality: Beside the image-to-term correlation for captioning, does MAGIC capture any-to-any medium correlations?

Our results show that MAGIC successfully exploits the image-to-term correlation to caption test images. Moreover, MAGIC is flexible and can caption multiple images as a group. We call this operation "*group captioning*" and present some qualitative results.

We also examine MAGIC's performance on spotting other cross-modal correlations. In particular, we show that MAGIC can capture same-modal correlations such as the term-term correlations: E.g., "given a term such as 'sky', find other terms that are likely to correspond to it." Potentially, MAGIC is also capable of spotting other correlations such as the reverse captioning problem: E.g., "given a term such as 'sky', find the regions that are likely to correspond to it." In general, MAGIC can capture any-to-any medium correlations.

4.1 Data set and G_{MAGIC} graph construction

Given a collection of captioned images $\mathcal{I}_{core}$, how do we select caption words for an uncaptioned image I_{new}? For automatic image captioning, we propose to caption I_{new} using the correlations between caption words and images in $\mathcal{I}_{core}$.

In our experiments, we use the same 10 sets of images from Corel that are also used in previous work [10, 3], so that our results can be compared to the previous results. In the following, the 10 captioned image sets are referred to as the "001", "002", ..., "010" sets. Each of the 10 data sets has around 5,200 images, and each image has about 4 caption words. These images are also called the *core images* from which we try to detect the correlations. For evaluation, accompanying each data set, a non-overlapping test set $\mathcal{I}_{test}$ of around 1,750 images is used for testing the captioning performance. Each test image has its ground truth caption.

Similar to previous work [10, 3], each image is represented by a set of image regions. Image regions are extracted using a standard segmentation tool [38], and each region is represented as a 30-D feature vector. The regional features include the mean and standard deviation of RGB values, average responses to various texture filters, its position in the entire image layout, and some shape descriptors (e.g., major orientation and the area ratio of bounding region to the real region). The image content is represented as a set-valued attribute "regions". In our experiments, an image has 10 regions on average. Figure 1(d,e,f) show some examples of image regions.

The exact region segmentation and feature extraction details are *orthogonal* to our approach - any published segmentation methods and feature extraction functions [12] will suffice. All our MAGIC method needs is a black box that will map each color image into a set of zero or more feature vectors.

We want to stress that there is no given information about which region is associated with which term in the core image set - all we know is that a set of regions co-occurs with a set of terms in an image. That is, no alignment information between individual regions and terms is available.

Therefore, a captioned image becomes an object with two set-valued attributes: "regions" and "terms". Since the regions and terms of an image are correlated, we propose to use MAGIC to detect this correlation and use it to predict the missing caption terms correlated with the uncaptioned test images.

The first step of MAGIC is to construct the MAGIC graph. Following the instructions for graph construction in Section 3.1, the graph for captioned images with attributes "regions" and "terms" will be a 3-layer graph with nodes for images, regions and terms. To form the NN-links, we define the distance function (Assumption 1) between two regions (tokens) as the L_2 norm between their feature vectors. Also, we define that two terms are similar if and only if they are identical, i.e., no term is any other's neighbor. As a result, there is no NN-link between term nodes.

For results shown in this section, the number of nearest neighbors between attribute/domain tokens is k=3. However, as we will show later in Section 5.1, the captioning accuracy is insensitive to the choice of k. In total, each data set has about 50,000 different region tokens and 160 words, resulting in a graph G_{MAGIC} with about 55,500 nodes and 180,000 edges. The graph based on the core image set $\mathcal{I}_{core}$ captures the correlations between regions and terms. We call such graph the *"core" graph.*

How do we caption a new image, using the information in a MAGIC graph?' Similar to the core images, an uncaptioned image I_{new} is also an object with set-valued attributes: "regions" and "caption", where attribute "caption" has null value. To find caption words correlated with image I_{new}, we propose to look at regions in the core image set that are similar to the regions of I_{new}, and find the words that are correlated with these core image regions. Therefore, our algorithm has two main steps: finding similar regions in the core image set (augmentation) and identifying caption words (RWR). Next, we define "core graph", "augmentation" and "gateway nodes", to facilitate the description of our algorithm.

Definition 4 (Core graph, augmentation and gateway nodes) *For automatic image captioning, we define the* ***core*** *of the G_{MAGIC}, G_{core}, be the subgraph that constitutes information in the given captioned images $\mathcal{I}_{core}$. The graph G_{MAGIC} for captioning a test image I_{new} is an* ***augmented graph****, which is the core G_{core} augmented with the region-nodes and image-node of I_{new}. The augmentation subgraph is denoted as G_{aug}, and hence the overall G_{MAGIC}=$G_{core} \cup G_{aug}$. The nodes in the core subgraph G_{core} that are adjacent to the augmentation are called the* ***gateway*** *nodes, $\mathcal{GW}$.*

As an illustration, Figure 2 shows the graph G_{MAGIC} for two core (captioned) images $\mathcal{I}_{core}$=$\{I_1, I_2\}$ and one test (to-be-captioned) image $\mathcal{I}_{test}$=$\{I_3\}$, with the parameter for NN-links k=1. The core subgraph G_{core} contains region nodes $\{r_1, \ldots, r_7\}$, image nodes $\{I_1, I_2\}$, and all the term nodes $\{t_1, \ldots, t_8\}$. The augmentation G_{aug} contains region nodes $\{r_8, \ldots, r_{11}\}$ and the image node $\{I_3\}$ of the test image. The gateway nodes are the region nodes $\mathcal{GW}$=$\{r_5, r_6, r_7\}$ that bridge the G_{core} and G_{aug}.

Different test images have different augmented graphs and gateway nodes. However, since we will caption only one test image at a time, the symbols G_{aug} and $\mathcal{GW}$ represent for the augmented graph and gateway nodes of the test image in question.

The first step of our image captioning algorithm, augmentation, can be done by finding the gateway nodes - the collection of the k nearest neighbors of each region of I_{new}. In the second step, we propose to use RWR, restarting from the test image-node, to identify the correlated words (term-nodes). A predicted caption of g words for the image I_{new} will correspond to the g term-nodes with highest RWR scores. Figure 6 gives the details of our algorithm.

To sum up, for image captioning, the core of the G_{MAGIC} is first constructed based on the given captioned images $\mathcal{I}_{core}$. Then, each test image I_{new} is captioned, one by one, in steps summarized in Figure 6.

4.2 Captioning accuracy

We measure captioning performance by the captioning accuracy, which is defined as the fraction of terms which are correctly predicted. Following the same evaluation procedure as that in previous

Input: 1. The core graph G_{core}, an image I_{new} to be captioned, and

2. g, the number of caption words we want to predict for I_{new}.

Output: Predicted caption words for I_{new}.

Steps:

1. Augment the image node and region nodes of I_{new} to the core graph G_{core}.
2. Do RWR from the image node of I_{new} on the augmented graph G_{MAGIC} (Algorithm 4).
3. Rank all term nodes by their RWR scores.
4. The g top-ranked terms will be the output - the predicted caption for I_{new}.

Figure 6: Steps to caption an image, using the proposed MAGIC framework.

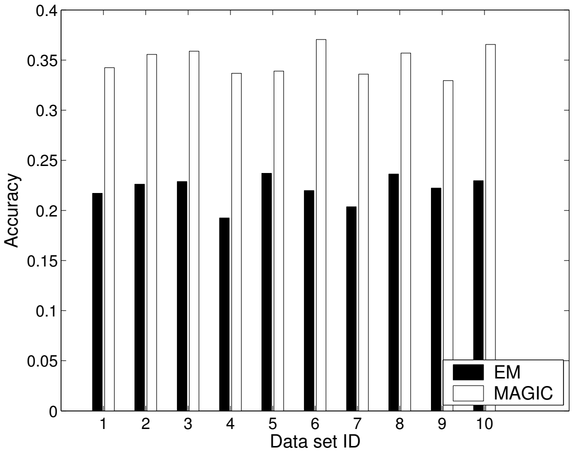


Figure 7: Comparing MAGIC to the EM method. The parameters for MAGIC are $c = 0.66$ and $k = 3$. The x-axis shows the 10 data sets, and the y-axis is the average captioning accuracy over all test images in a set.

work [10, 3], for a test image which has g ground-truth caption terms, MAGIC will also predict g terms. If p of the predicted terms are correct, then the captioning accuracy acc on this test image is defined as

$$acc = \frac{p}{g}.$$

The average captioning accuracy $\overline{acc}$ on a set of T test images is defined as

$$\overline{acc} = \frac{1}{T}\sum_{i=1}^{T} acc_i,$$

where acc_i is the captioning accuracy on the i-th test image.

Figure 7 shows the average captioning accuracy on the 10 image sets. We compare our results with those reported in [10]. The method in [10] is one of the most recent and sophisticated: it

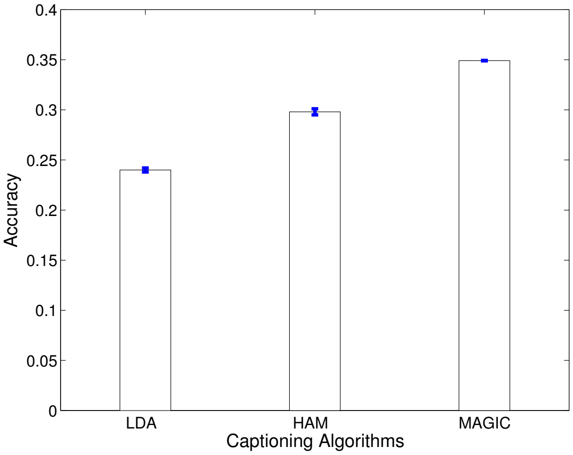


Figure 8: Comparing MAGIC with LDA and HAM. The mean and variance of the average accuracy over the 10 Corel data sets are shown at the y-axis - LDA: (μ, σ^2)=(0.24,0.002); HAM: (μ, σ^2)=(0.298,0.003); MAGIC : (μ, σ^2)=(0.3503, 0.0002). μ: mean average accuracy. σ^2: variance of average accuracy. The length of the error bars at the top of each bar is 2σ.

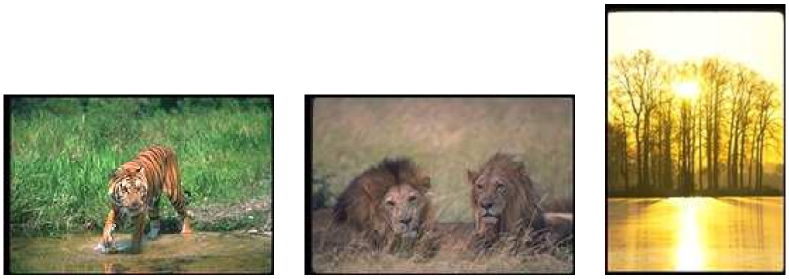
models the image captioning problem as a statistical translation modeling problem and solves it using expectation-maximization (EM). We refer to their method as the "EM" approach. The x-axis groups the performance numbers of MAGIC (white bars) and EM (black bars) on the 10 data sets. On average, MAGIC achieves captioning accuracy improvement of 12.9 percentage points over the EM approach, which corresponds to a relative improvement of 58%.

We also compare the captioning accuracy with even more recent machine vision methods [3], on the same data sets: the Hierarchical Aspect Models method ("HAM"), and the Latent Dirichlet Allocation model ("LDA"). Figure 8 compares MAGIC with LDA and HAM, in terms of the mean and variance of the average captioning accuracy over the 10 data sets. Although both HAM and LDA improve on the EM method, they both lose to our generic MAGIC approach (35%, versus 29% and 25%). It is also interesting that MAGIC gives significantly lower variance, by roughly an order of magnitude: 0.002 versus 0.02 and 0.03. A lower variance indicates that the proposed MAGIC method is more robust to variations among different data sets.

Figure 9 shows some examples of the captions given by MAGIC. For the test image I_3 in Figure 1, MAGIC captions it correctly (Figure 9(a)). In Figure 9(b), MAGIC surprisingly gets the word "mane" correctly; however, it mixes up "buildings" with "tree" (Figure 9(c)).

4.3 Generalization

MAGIC treats information from all media uniformly as nodes in a graph. Since all nodes are basically the same, we can do RWR and restart from any subset of nodes of any medium, to detect



	(a)	(b)	(c)
Truth	cat, grass, tiger, water	mane, cat, lion, grass	sun, water, tree, sky
MAGIC	grass, cat, tiger, water	lion, grass, cat, mane	tree, water, buildings, sky

Figure 9: Terms are ordered by their given importance. Figures look best in color.

any-to-any medium correlations. The flexibility of our graph-based framework also enables novel applications, such as captioning images in groups (*group captioning*). In this subsection, we show results on (a) spotting the term-to-term correlation in image captioning data sets, and (b) group captioning.

Beyond image-to-term correlation MAGIC successfully exploits the image-to-term correlation for captioning images. However, the MAGIC graph G_{MAGIC} contains correlations between all media (image, region, and term). To show how well MAGIC works on objects of any medium, we design an experiment to identify correlated captioning terms, using the term-to-term correlation in the graph G_{MAGIC}.

We use the same 3-layer MAGIC core graph G_{core} that was constructed in the previous subsection for automatic image captioning. Given a query term t, we use RWR to find other terms correlated with it. Specifically, we perform RWR, restarting from the query term(-node). The terms deemed correlated with the query term are term(-node)s that receive high RWR scores.

Table 2 shows the top 5 terms with the highest RWR scores for some query terms. In the table, each row shows the query term at the first column, followed by the top 5 correlated terms selected by MAGIC (sorted by their RWR scores). The selected terms make a lot of sense, and have meanings related with the query term. For example, the term "branch", when used in image captions, is strongly related to forest- or bird- related concepts. MAGIC shows exactly this, correlating "branch" with terms such as "birds", "owl" and "nest".

A second, subtle observation, is that our method does not seem to be biased by frequent words. In our collection, the terms "water" and "sky" are more frequent than the others (like the terms "the" and "a" in normal English text). Yet, these frequent terms do *not* show up too often in

Query term	1	2	3	4	5
branch	birds	night	owl	nest	hawk
bridge	water	arch	sky	stone	boats
cactus	saguaro	desert	sky	grass	sunset
car	tracks	street	buildings	turn	prototype
f-16	plane	jet	sky	runway	water
market	people	street	food	closeup	buildings
mushrooms	fungus	ground	tree	plants	coral
pillars	stone	temple	people	sculpture	ruins
reefs	fish	water	ocean	coral	sea
textile	pattern	background	texture	designs	close-up

Table 2: Correlated terms of some query terms

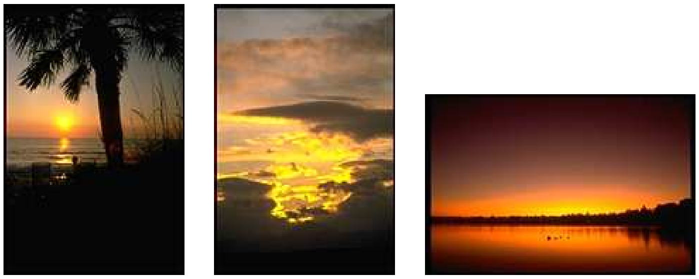
Table 2, as a correlated term of a query term. It is surprising, given that we did nothing special when using MAGIC: no tf/idf weighting, no normalization, and no other domain-specific analysis. We just treated these frequent terms as nodes in our MAGIC graph, like any other nodes.

Group captioning The proposed MAGIC method can be easily extended to caption a group of images, considering all of them at once. This flexibility is due to the graph-based framework of MAGIC, which allows augmentation of multiple nodes and doing RWR from any subset of nodes. To the best of our knowledge, MAGIC is the first method that is capable of doing group captioning.

Application 2 (Group captioning) *Given a set $\mathcal{I}_{core}$ of color images, each with caption words; and given a (query) group of uncaptioned images $\{I'_1, \ldots, I'_t\}$, find the best g (say, g=5) caption words to assign to the group.*

Possible applications for group captioning include video segment captioning, where a video segment is captioned according to the group of keyframes associated with the segment. Since keyframes in a segment are related, captioning them as a whole can take into account the inter-keyframe correlations, which are missed if each keyframe is captioned separately. Accurate captions for video segments may improve performances on tasks such as video retrieval and classification.

The steps to caption a group of images are similar to those for the single-image captioning outlined in Figure 6. A core MAGIC graph is still used to capture the mixed media information of the given collection of images. The differences for group captioning are, instead of augmenting the single-image to the core and restarting from it, now we augment all t images in the query group $\{I'_1, \ldots, I'_t\}$ to the core, and restarts randomly from one of the images in the group (i.e., each with probability $1/t$ to be the restart node).



	(a)	(b)	(c)
Truth	sun, water, tree, sky	sun, clouds, sky, horizon	sun, water
MAGIC	tree, people, sky, water	water, tree, people, sky	sky, sun
Group		sky, water, tree, sun	

Figure 10: Group captioning: Captioning terms with highest RWR scores are listed first.

Figure 10 shows the result of using MAGIC for captioning a group of three images. MAGIC found reasonable terms for the entire group of images: “sky”, “water”, “tree”, and “sun”. Captioning multiple images as a group takes into consideration the correlations between different images in the group, and in this example, this helps reduce the scores of irrelevant terms such as “people”. In contrast, when we caption these images individually, MAGIC selects “people” as caption words for images in Figure 10(a) and (b), which do not contain people-related objects.

5 System Issues

MAGIC provides an intuitive framework for detecting cross-modal correlations. The RWR computation in MAGIC is fast that it scales linearly with the graph size. For example, a straightforward implementation of RWR can caption an image in less than 5 seconds.

In this section, we discuss system issues such as parameter configuration and fast computation. In particular, we present results showing

- MAGIC is insensitive to parameter settings, and
- MAGIC is modular that we can easily employ the best module to date to speedup MAGIC.

5.1 Optimization of parameters

There are several design decisions to be made when employing MAGIC for correlation detection: *what should be the values for the two parameters: the number of neighbors k of a domain token,*

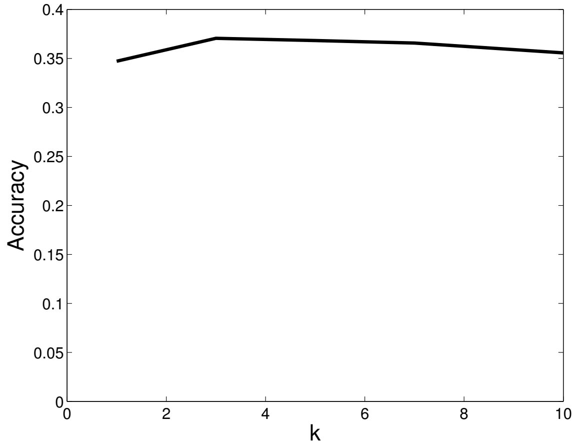


Figure 11: The plateau in the plot shows that the captioning accuracy is insensitive to value of the number of nearest neighbors k. Y-axis: Average accuracy over all images of data set "006". The restart probability is c=0.66.

and the restart probability c of RWR? And, *should we assign weights to edges, according to the types of their end points?* In this subsection, we empirically show that the performance of MAGIC is insensitive to these settings, and provide suggestions on determining reasonable default values.

We use automatic image captioning as the application to measure the effect of these parameters. The experiments in this section are performed on the same 10 captioned image sets ("001", ..., "010") described in Section 4.1, and we measure how the values of these parameters effect the captioning accuracy.

Number of Neighbors k The parameter k specifies the number of nearest domain tokens to which a domain token connects via the NN-links (Section 3.1). With these NN-links, objects having little difference in attribute values will be closer to each other in the graph, and therefore, are deemed more correlated by MAGIC. For k=0, all domain tokens are considered distinct; for larger k, our application is more tolerant to the difference in attribute values.

We examine the effect of various k values on image captioning accuracy. Figure 11 shows the captioning accuracy on the data set "006", with the restart probability c=0.66. The captioning accuracy increases as k increases from k=1, and reaches a plateau between k=3 and 10. The plateau indicates that MAGIC is insensitive to the value of k. Results on other data sets are similar, showing a plateau between k=3 and 10.

In hindsight, with only k=1, the collection of regions (domain tokens) is barely connected, missing important connections and thus leading to poor performance on detecting correlations. At the other extreme, with a high value of k, everybody is directly connected to everybody else, and there is no clear distinction between really close neighbors or just neighbors. For a medium number

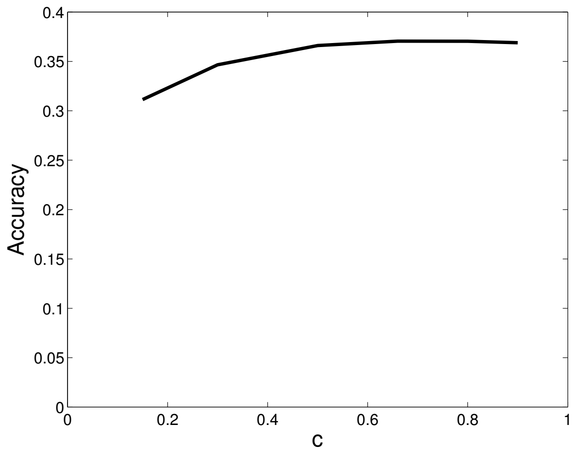


Figure 12: The plateau in the plot shows that the captioning accuracy is insensitive to value of the restart probability c. Y-axis: Average accuracy over all images of data set "006". The number of nearest neighbors per domain token is k=3.

of neighbors k, the NN-links apparently capture the correlations between the close neighbors, and avoid noise from remote neighbors. Small deviations from that value make little difference, which is probably because that the extra neighbors we add (when k increases), or those we retained (when k decreases), are at least as good as the previous ones.

Restart probability c The restart probability c specifies the probability to jump back to the restarting node(s) of the random walk. Higher value of c implies giving higher RWR scores to nodes closer in the neighborhood of the restart node(s).

Figure 12 shows the image captioning accuracy of MAGIC with different values of c. The data set is "006", with the parameter k=3. The accuracy reaches a plateau between c=0.5 and 0.9, showing that the proposed MAGIC method is insensitive to the value of c. Results on other data sets are similar, showing a plateau between c=0.5 and 0.9.

For web graphs, the recommended value for c is typically c=0.15 [14]. Surprisingly, our experiments show that this choice does not give good performance. Instead, good quality is achieved for c=0.6 $\sim$ 0.9. Why is this discrepancy?

We conjecture that what determines a good value for the restart probability is the diameter of the graph. Ideally, we want our random walker to have a non-trivial chance to reach the outskirts of the whole graph. If the diameter of the graph is d, the probability that the random walker (with restarts) will reach a point on the periphery is proportional to $(1 - c)^d$.

For the web graph, the diameter is estimated to be d=19 [1]. This implies that the probability

	w_{region}		
w_{term}	0.1	1	10
0.1	0.370332	0.371963	0.370812
1	0.369900	0.370524	0.371963
10	0.368969	0.369181	0.369948

Table 3: Captioning accuracy is insensitive to various weight settings on OAV-links to the two media: region (w_{region}) and term (w_{term}).

$p_{periphery}$ for the random walker to reach a node in the periphery of the web graph is roughly

$$p_{periphery} = (1 - c)^{19} = (1 - 0.15)^{1}9 = 0.045 \ . \tag{2}$$

In our image captioning experiments, we use graphs that have three layers of nodes (Figures 2). The diameter of such graphs is roughly d=3. If we demand the same $p_{periphery}$ as equation (2), then the c value for our 3-layer graph would be

$$\begin{align} (1 - 0.15)^{19} &= (1 - c)^{3} \tag{3} \\ \Rightarrow c &= 0.65 \ , \tag{4} \end{align}$$

which is much closer to our empirical observations. Of course, the problem requires more careful analysis - but we are the first to show that c=0.15 is not always optimal for random walk with restarts.

Link weights MAGIC uses a graph to encode the relationship between mixed media objects and their attributes of different media. The OAV-links in the graph connect objects to their domain tokens (Figure 2). To give more attention to an attribute domain D, we can increase the weights of OAV-links that connect to tokens of domain D. *Should we treat all media equally, or should we weight OAV-links according to their associated domains? How should we weight the OAV-links? Could we achieve better performance on weighted graphs?*

We investigate how the change on link weights influences image captioning accuracy. Table 3 shows the captioning accuracy on data set "006" when different weights are assigned on the OAV-links to regions (weight w_{region}) and those to terms (w_{term}). For all cases, the number of nearest neighbors is k=3 and the restart probability is c=0.66. The case where (w_{region}, w_{term})=(1,1) is that of the unweighted graph, and is the result we reported in Section 4. As link weights vary from 0.1, 1 to 10, the captioning accuracy is basically unaffected. The results on other data sets are similar - captioning accuracy is at the same level on a weighted graph as on the unweighted graph.

This experiment shows that an unweighted graph is appropriate for our image captioning application. We speculate that an appropriate weighting for an application depends on properties such as the number of attribute domains (i.e., the number of layers in the graph), the average size of a set-valued attribute of an object (such as, average number of regions per image), and so on. We plan to investigate more on this issue in our future work.

5.2 Speedup graph construction by approximation

The proposed MAGIC method encodes a mixed media data set as a graph, and employs the RWR algorithm to find cross-modal correlations. The construction of the G_{MAGIC} graph is intuitive and straightforward, and the RWR computation is light and linear to the data base size. One step which is relatively expensive is the construction of NN-links in a MAGIC graph.

When constructing the NN-links of a MAGIC graph, we need to compute the nearest neighbors for every domain token. For example, in our image captioning experiments (Section 4.1), to form the NN-links among region-nodes in the MAGIC graph, k-NN searches are performed 50,000 times (one for each region token) in the 30-dimensional region-feature space.

In MAGIC, the NN-links are proposed to capture the similarity relation among domain tokens. The goal is to associate tokens that are similar, and therefore, it could be suffice to have the NN-links connect to neighbors which are close enough, even if they are not exactly the closest ones. The approximate nearest neighbor search is usually faster, by trading accuracy for speed. The interesting questions are: *How much speedup could we gain by allowing approximate NN-links? How much is the performance reduction by approximation?*

For efficient nearest neighbor search, one common way is to use a spatial index such as R-tree [37], which give exact nearest neighbor in logarithmic time. Fortunately, MAGIC is modular and we can pick the best module to perform each step. In our experiments, we used the approximate nearest neighbor method (ANN) [2], which supports both exact and approximate nearest neighbor search. ANN estimates the distance to a nearest neighbor up to $(1+\epsilon)$ times the actual distance: $\epsilon = 0$ means exact search, no approximation; bigger ϵ values give rougher estimation.

Table 4 lists the average wall clock time to compute the top 10 neighbors of a region in the 10 Corel image sets of our image captioning experiments. Compared to the sequential search, the speedup of using a spatial method increases from 12.1 to 51.1, from exact search to a rough approximation of $epsilon$ = 0.8. For the top k=3 nearest neighbors (the setting used in our experiments), the error percentage is at most 0.46% for the roughest approximation, equivalent to making one error in every 217 NN-links. The sequential search method is implemented in C++, and is compiled with the code optimization (`g++ -O3`).

The small differences on NN-links do not change the characteristic of the MAGIC graph significantly, and has limited affect on the performance of image captioning. At ϵ=0.2, no error is made on the NN-links in the MAGIC graph, and therefore the captioning accuracy is the same as exact computation. At ϵ=0.8, the average captioning accuracy decreases by just 1.59 percentage point,

	ANN			Sequential search (SS)
	ϵ=0	ϵ=0.2	ϵ=0.8	
Elapse time (msec.)	3.8	2.4	0.9	46
Speedup to SS	12.1	19.2	51.1	1
Error (in top k=10)	-	0.0015%	1.67%	-
Error (in top k=3)	-	-	0.46%	-

Table 4: Computation/approximation trade off in the NN-link construction among image regions. The distance to a neighboring point is approximated to within (1+ϵ) times the actual distance. ϵ=0 indicates the exact k-NN computation. Elapse time: average wall clock time for one nearest neighbor search. Speedup: the ratio of elapse time, with respect to the time of sequential search (SS). Error: the percentage of mistakes made by approximation in the k nearest neighbors. The symbol "-" means zero error.

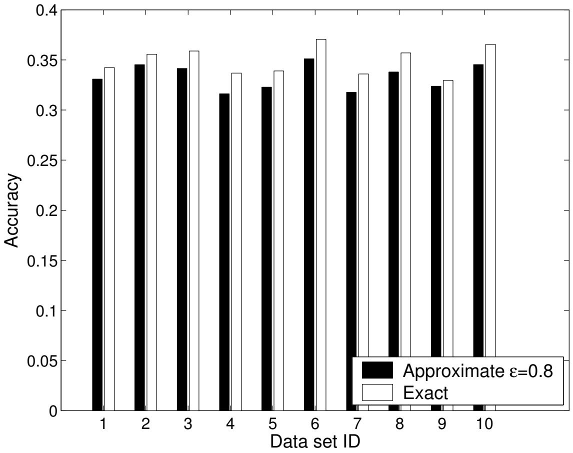


Figure 13: Using approximate NN-links ($epsilon$=0.8) reduces captioning accuracy by just 1.59% on the average. X-axis: 10 data sets. Y-axis: average captioning accuracy over test images in a set. The parameters for MAGIC are $c = 0.66$ and $k = 3$.

averaged over the 10 Corel image sets (Figure 13).

6 Conclusions

Mixed media objects such as captioned images or video clips contain attributes of different modalities (image, text, or audio). Correlations across different modalities provide information about the multimedia content, and are useful in applications ranging from summarization to semantic captioning. In this paper, we developed MAGIC, a graph-based method for detecting cross-modal correlations in mixed media data set.

There are two challenges in detecting cross-modal correlations, namely, representation of attributes of various modalities and the detection of correlations among any subset of modalities. MAGIC turns the multimedia problem into a graph problem, and provides an intuitive solution that easily incorporates various modalities. The graph framework of MAGIC creates opportunity for applying graph algorithms to multimedia problems. In particular, MAGIC finds cross-modal correlations using the technique of random walk with restarts (RWR), which accommodates set-valued attributes and data noise with no extra effort.

We applied MAGIC for automatic image captioning. By finding robust correlations between text and image, MAGIC achieved a relative improvement by 58% in captioning accuracy as compared to recent machine learning techniques (Figure 8). Moreover, the MAGIC framework enabled novel data mining applications, such as *group captioning* where multiple images are captioned simultaneously, taking into account the possible correlations between the multiple images in the group (Figure 10).

Technically, MAGIC has the following desirable characteristics:

- It is domain independent: The $Sim_i(*, *)$ similarity functions (Assumption 1) completely isolate our MAGIC method from the specifics of an application domain, and make MAGIC applicable to detect correlations in all kinds of mixed media data sets.

- It requires no fine-tuning on parameters or link weights: The performance is not sensitive to the two parameters - the number of neighbors k and the restart probability c, and it requires no special weighting scheme like tf/idf for link weights (Section 5.1).

- It is fast and scales up well with the database/graph size.

- It is modular and can easily tap recent advances in related areas (e.g., fast nearest neighbor search) to improve performance (Section 5.2).

We are pleasantly surprised that such a domain-independent method, with no parameters to tune, outperformed some of the most recent and carefully tuned methods for automatic image captioning. Most of all, the graph-based framework proposed by MAGIC creates opportunity for applying graph algorithms to multimedia problems. Future work could further exploit the promising connection between multimedia databases and graph algorithms, including multi-modal event summarization [32, 31], outlier detection, and other data mining task that require the discovery of correlations as its first step.

References

[1] A. Albert, H. Jeong, and A.-L. Barabasi. Diameter of the world wide web. *Nature*, 401:130–131, 1999.

[2] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, and A. Wu. An optimal algorithm for approximate nearest neighbor searching. *Journal of the ACM*, (45):891–923, 1998.

[3] K. Barnard, P. Duygulu, N. de Freitas, D. A. Forsyth, D. Blei, and M. Jordan. Matching words and pictures. *Journal of Machine Learning Research*, 3:1107–1135, 2003.

[4] A. B. Benitez and S.-F. Chang. Multimedia knowledge integration, summarization and evaluation. In *Proceedings of the 2002 International Workshop on Multimedia Data Mining in conjuction with the International Conference on Knowledge Discovery and Data Mining (MDM/KDD-2002), Edmonton, Alberta, Canada, July 23-26*, 2002.

[5] D. Blei and M. I. Jordan. Modeling annotated data. In *Proceedings of the 26th ACM SIGIR Conference*, July 28-August 1, 2003, Toronto, Canada.

[6] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *Proceedings of the Seventh International World Wide Web Conference*, 1998.

[7] S.-F. Chang, R. Manmatha, and T.-S. Chua. Combining text and audio-visual features in video indexing. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2005), Philadelphia, PA*, March 2005.

[8] A. P. de Vries, T. Westerveld, and T. Ianeva. Combining multiple representations on the TRECVID search task. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '04)*, volume 3, pages 1052–1055, May 2004.

[9] P. G. Doyle and J. L. Snell. *Random Walks and Electric Networks*, volume 22. The Mathematical Association of America, 1984.

[10] P. Duygulu, K. Barnard, N. Freitas, and D. A. Forsyth. Object recognition as machine translation: learning a lexicon for a fixed image vocabulary. In *Seventh European Conference on Computer Vision (ECCV)*, volume 4, pages 97–112, 2002.

[11] P. Duygulu, J.-Y. Pan, and D. A. Forsyth. Towards auto-documentary: Tracking the evolution of news stories. In *Proceedings of the ACM Multimedia Conference*, October 2004.

[12] C. Faloutsos. *Searching Multimedia Databases by Content.* Number 3 in The Kluwer International Series On Advances In Database Systems. Kluwer Academic Publishers Group, The Netherlands, August 1996.

[13] S. L. Feng, R. Manmatha, and V. Lavrenko. Multiple bernoulli relevance models for image and video annotation. In *Proceedings of the International Conference on Pattern Recognition (CVPR 2004)*, volume 2, pages 1002–1009, June 2004.

[14] T. Haveliwala, S. Kamvar, and G. Jeh. An analytical comparison of approaches to personalizing pagerank. Technical Report 2003-35, Stanford University, June 2003. http://dbpubs.stanford.edu/pub/2003-35, 20th of June 2003, 4 pages.

[15] T. H. Haveliwala. Topic-sensitive pagerank. In *WWW2002*, May 7-11 2002.

[16] W. Hsu, L. Kennedy, C.-W. Huang, S.-F. Chang, C.-Y. Lin, and G. Iyengar. News video story segmentation using fusion of multi-level multi-modal features in TRECVID 2003. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2004), Montreal, Canada*, May 2004.

[17] J. Jeon, V. Lavrenko, and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models. In *26th Annual International ACM SIGIR Conference*, July 28-August 1, 2003, Toronto, Canada.

[18] R. Jin, J. Y. Chai, and L. Si. Effective automatic image annotation via a coherent language model and active learning. In *Proceedings of the 12th annual ACM international conference on Multimedia, New York, NY, USA*, pages 892 – 899, October 2004.

[19] S. D. Kamvar, T. H. Haveliwala, and G. H. Golub. Adaptive methods for the computation of pagerank. In *Proceedings of the International Conference on the Numerical Solution of Markov Chains (NSMC)*, September 2003.

[20] S. D. Kamvar, T. H. Haveliwala, C. D. Manning, and G. H. Golub. Extrapolation methods for accelerating pagerank computation. In *Proceedings of the 12th World Wide Web Conference*, 2003.

[21] J. Kleinberg. Authoritative sources in a hyperlinked environment. In *Proceedings of the 9th ACM-SIAM Symposium on Discrete Algorithms*, 1998.

[22] D. Li, N. Dimitrova, M. Li, and I. K. Sethi. Multimedia content processing through cross-modal association. In *Proceedings of the eleventh ACM international conference on Multimedia, Berkeley, CA, USA*, pages 604–611, 2003.

[23] J. Li and J. Z. Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(10):14, 2003.

[24] W.-H. Lin and A. Hauptmann. News video classification using svm-based multimodal classifiers and combination strategies. In *Proceedings of the 10th annual ACM international conference on Multimedia, Juan Les Pins, France*, October 2002.

[25] L. Lovasz. Random walks on graphs: A survey. *Combinatorics, Paul Erdos is Eighty*, 2:353–398, 1996.

[26] O. Maron and A. L. Ratan. Multiple-instance learning for natural scene classification. In *The Fifteenth International Conference on Machine Learning*, 1998.

[27] Y. Mori, H. Takahashi, and R. Oka. Image-to-word transformation based on dividing and vector quantizing images with words. In *First International Workshop on Multimedia Intelligent Storage and Retrieval Management*, 1999.

[28] M. R. Naphade, I. Kozintsev, and T. Huang. Probabilistic semantic video indexing. In *Advances in Neural Information Processing Systems 13, Papers from Neural Information Processing Systems (NIPS) 2000, Denver, CO, USA*, 2001.

[29] C. R. Palmer and C. Faloutsos. Electricity based external similarity of categorical attributes. In *PAKDD 2003*, May 2003.

[30] J.-Y. Pan and C. Faloutsos. VideoCube: a novel tool for video mining and classification. In *Proceedings of the Fifth International Conference on Asian Digital Libraries (ICADL 2002)*, 2002.

[31] J.-Y. Pan, H. Yang, and C. Faloutsos. MMSS: Graph-based multi-modal story-oriented video summarization and retrieval. Technical report, CMU-CS-04-114, Carnegie Mellon University, 2004.

[32] J.-Y. Pan, H. Yang, and C. Faloutsos. MMSS: Multi-modal story-oriented video summarization. In *Proceedings of the Fourth IEEE International Conference on Data Mining (ICDM 04)*, 2004.

[33] J.-Y. Pan, H.-J. Yang, P. Duygulu, and C. Faloutsos. Automatic image captioning. In *Proceedings of the 2004 IEEE International Conference on Multimedia and Expo (ICME 2004), Taipei, Taiwan*, June 2004.

[34] J.-Y. Pan, H.-J. Yang, C. Faloutsos, and P. Duygulu. Automatic multimedia cross-modal correlation discovery. In *Proceedings of the 10th ACM SIGKDD Conference*, August 2004.

[35] J.-Y. Pan, H.-J. Yang, C. Faloutsos, and P. Duygulu. GCap: Graph-based automatic image captioning. In *Proceedings of the 4th International Workshop on Multimedia Data and Document Engineering (MDDE-04)*, July 2, 2004 2004.

[36] N. Sebe, M. S. Lew, X. Zhou, T. Huang, and E. Bakker. The state of the art in image and video retrieval. In *Proceedings of the International Conference on Image and Video Retrieval (CIVR'03)*, pages 1–8, July 2003.

[37] T. K. Sellis, N. Roussopoulos, and C. Faloutsos. The R+-tree: A dynamic index for multi-dimensional objects. In *Proceedings of the 12th International Conference on VLDB*, pages 507–518, September 1987.

[38] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[39] R. K. Srihari, A. Rao, B. Han, S. Munirathnam, and X. Wu. A model for multimodal information retrieval. In *Proceedings of the IEEE International Conference on Multimedia and Expo, 2000 (ICME 2000)*, volume 2, pages 701–704, July 2000.

[40] P. Virga and P. Duygulu. Systematic evaluation of machine translation methods for image and video annotation. In *Proceedings of The Fourth International Conference on Image and Video Retrieval (CIVR 2005), Singapore*, July 2005.

[41] X.-J. Wang, W.-Y. Ma, G.-R. Xue, and X. Li. Multi-model similarity propagation and its application for web image retrieval. In *Proceedings of the 12th annual ACM international conference on Multimedia, New York, NY, USA*, pages 944–951, October 2004.

[42] L. Wenyin, S. Dumais, Y. Sun, H. Zhang, M. Czerwinski, and B. Field. Semi-automatic image annotation. In *INTERACT2001, 8th IFIP TC.13 Conference on Human-Computer Interacti on*, Tokyo, Japan July 9-13, 2001.

[43] Y. Wu, E. Y. Chang, K. C.-C. Chang, and J. R. Smith. Optimal multimodal fusion for multimedia data analysis. In *Proceedings of the 12th annual ACM international conference on Multimedia, New York, NY, USA*, pages 572–579, October 2004.

[44] L. Xie, L. Kennedy, S.-F. Chang, A. Divakaran, H. Sun, and C.-Y. Lin. Layered dynamic mixture model for pattern discovery in asynchronous multi-modal streams. In *IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP 2005), Philadelphia*, March 2005.

[45] D.-Q. Zhang, C.-Y. Lin, S.-F. Chang, and J. R. Smith. Semantic video clustering across sources using bipartite spectral clustering. In *Proceeding of IEEE Conference of Multimedia and Expo, 2004 (ICME 2004)*, June 2004.

[46] Z. Zhang, R. Zhang, and J. Ohya. Exploiting the cognitive synergy between different media modalities in multimodal information retrieval. In *Proceedings of the IEEE International Conference on Multimedia and Expo, 2004 (ICME 2004)*, volume 3, pages 2227–2230, June 2004.

[47] R. Zhao and W. I. Grosky. Bridging the semantic gap in image retrieval. In T. K. Shih, editor, *Distributed Multimedia Databases: Techniques and Applications*, pages 14–36. Idea Group Publishing, Hershey, Pennsylvania, 2001.