Tripartite Graph Models for Multi Modal Image Retrieval

Chandrika Pulla
chandrika@research.iiit.ac.in
C.V.Jawahar
jawahar@iiit.ac.in

Most of the traditional image retrieval methods use either low level visual features or embedded text for representation and indexing. In recent years, there has been significant interest in combining these two different modalities for effective retrieval. In this paper, we propose a tri-partite graph based representation of the multi model data for image retrieval tasks. Our representation is ideally suited for dynamically changing or evolving data sets, where repeated semantic indexing is practically impossible. An undirected tripartite graph \( G = (T, V, D, E) \) has three sets of vertices where, \( T = \{ t_1, t_2, \ldots, t_n \} \) are text words, \( V = \{ v_1, v_2, \ldots, v_m \} \) are visual words and \( D = \{ d_1, d_2, \ldots, d_i \} \) are images with \( E = \{ e_{d_1}, e_{d_2}, e_{d_3}, \ldots, e_{d_m}, e_{i_1}, \ldots, e_{i_n} \} \) as set of edges. Figure 1 pictorially represent the tripartite graph model (TGM) we use.

![Tripartite Graph Representation of data set, \( t_{w_i} \) are text words, \( v_{w_i} \) are visual words and \( d_i \) are the images](image)

Figure 1: Tri-partite Graph Representation of data set, \( t_{w_i} \) are text words, \( v_{w_i} \) are visual words and \( d_i \) are the images

Thus this model has three sets of vertices (images, text words and visual words) and edges going from one set to other. The nodes correspond to visual words as well as text words store the inverse document frequency (IDF) corresponding to the document/image collection. The edges from text words to images as well as those from visual words to images, encode the term frequency (TF) corresponding to the word-image pair. However, the weights of edges which relate the text words with visual words cannot be directly assigned. These edges are weighed as:

\[
W_{pq} = \frac{\sum C_{t_p, v_q}(ae_{d_i} + (1 - a)e_{i_q})}{\sum ae_{d_i} + (1 - a)e_{i_q}}
\]

Where \( C_{t_p, v_q} = 1 \), if \( t_p \) and \( v_q \) are there in document \( d_i \). Since the documents (images) are the entity which connects text words and visual words, summations are carried out over the images/documents.

For indexing, a tripartite graph \( G \) is constructed with the nodes and edges as mentioned above. Given a collection of images and textual tags, building a TGM is possible. However, when additional images come, TGM shows its advantage in insertion. To insert an additional image, the edge weights of the tripartite graph to improve the retrieval performance. The Figure 2 shows the comparison of these methods in mean Average Precision(mAP) values. We used four data sets, University of Washington(UW) Data set [4], Multi-label Image Data set [6], IAPR TC12 Data set [3], NUS-WIDE [2] for the evaluation of the methods proposed. For all our experiments the number of concepts is determined by the concepts present in the respective databases that are known. The mAP results show that performance of TGM is comparable to other methods. The performance of TGM with weighted-learning is slightly better that with the TF. The advantage of TGM is noticeable when new images are added to database. TGM takes only few milliseconds for semantic indexing whereas for variants of pLSA the entire semantic indexing needs to be done again, incurring high time and memory costs.

![Algorithm 1 Graph Partitioning Algorithm for Tripartite Graph](image)

Algorithm 1 Graph Partitioning Algorithm for Tripartite Graph

\[
\text{def } \text{GP}(G, N, R)
\]

Update amount of Relevance score \( R \) that have passed through node \( N \)

\[
R[N] \leftarrow R
\]

if Node \( N \) is of type Word

\[
R = R \cdot \text{IDF}(N_i) \text{ is propagated to document.}
\]

end if

if Amount of labels transferable from \( N < \text{cutoff} \) then

exit

end if

for each node in neighbourhood of \( N \) do

\[
\text{GP}(G, \text{node}, R \cdot \text{TF}(N, \text{node}))
\]

end for


