Multi-modal retrieval using binary hashcodes

- **Research Question**: Can learning binary hashcodes for cross-modal data-points enable efficient and effective multi-modal retrieval?
- **Approach**: we learn hyperplanes that split both feature spaces (e.g. text, image descriptors) into buckets so that similar cross-modal data-points fall into the buckets labelled with the same hashcodes.

Hashing-based approximate nearest neighbour search

- **Problem**: Nearest Neighbour (NN) search in multi-modal datasets.
- **Hashing-based approach**:
  - Generate a similarity preserving binary hashcode for query.
  - Use the fingerprint as index into the buckets of a hash table.
  - If collision occurs only compare to items in the same bucket.

- **Hashtable buckets are the polytopes formed by intersecting hyperplanes in the word and image descriptor feature spaces.**
- **This work**: learn hyperplanes to encourage collisions between similar multi-modal data-points.

Regularised Cross-Modal Hashing (RCMH)

- **Step A**: Use LSH [1] to initialise word bits \( B \in \{ -1, 1 \}^{N \times K} \)
  - \( N \): # data-points, \( K \): # bits
- **Repeat for \( M \) iterations**:
  - **Step B**: Graph regularisation, update the bits of each data-point to be the average of its neighbours:
    \[
    B \leftarrow \text{sgn} \left( \alpha \text{SD}^{-1} B + (1-\alpha) B \right)
    \]
  - \( * \in \{ 0, 1 \}^{N \times N} \): adjacency matrix, \( D \in \mathbb{Z}_+^{N \times N} \) diagonal degree matrix, \( B \in \{ -1, 1 \}^{N \times K} \): bits, \( \alpha \in [0, 1] \): interpolation parameter, \( \text{sgn} \): sign function
  - **Step C**: Word data-space partitioning, learn hyperplanes that predict the \( K \) word bits with maximum margin
    \[
    \min \| f_i \|^2 + C \sum_{i=1}^N \xi_{ik} \quad \text{s.t.} \quad B_{ik} (f_i^T a_i + b_k) \geq 1 - \xi_{ik} \quad \text{for} \ i = 1 \ldots N
    \]
  - \( f_i \in \mathbb{R}^D \): word hyperplane, \( b_k \in \mathbb{R} \): bias, \( a_i \in \mathbb{R}^D \): word descriptor, \( B_{ik} \): bit for word data-point \( a_i \), \( \xi_{ik} \): slack variable
  - **Step D**: Update \( B \):
    \[
    b_k \leftarrow \text{sgn} (f_i^T a_i + b_k)
    \]
  - Use the learnt hyperplanes \( \{ f_i, g_k \}_{k=1}^K \) to generate hashcodes

Step B: Graph Regularisation (bit smoothing)

- Set word bits of each data-point to be the average of its neighbours:

Step C: Learning the hashtable buckets (hyperplanes)

- Word hyperplanes \( f_1, f_2 \) learnt using bit 1 (green), bit 2 (red) as labels:
- Visual hyperplanes \( g_1, g_2 \) learnt using same regularised bits:

Results: Retrieval effectiveness and efficiency

- Retrieval evaluation on two standard multi-modal (text,image) datasets. Image query used to retrieve documents, and vice-versa.
- Retrieval on NUS-WIDE (left). Timing on Wiki dataset (right).

Conclusions and References

- New effective and efficient iterative model for cross-modal hashing
- Hashcode bits smoothed over using adjacency graph are used to learn hashable buckets (hyperplanes) in word and image space.
- Extend to high volume data stream and cross-lingual retrieval.

References


Our model more effective and efficient than competitive baselines.