A Selective Survey on Multimodal Representation, Fusion and Retrieval

SHIVANGI BITHEL*, DWIJESH GOHIL*, and PRABHAT KANAUJIA*, Indian Institute of Technology, New Delhi, India

Humans use numerous communicative modalities, including linguistic, acoustic, and visual. Multimodal Information Retrieval tries to combine these multiple modalities for building a robust ranking system. This survey summarizes Representation Learning, Fusion strategies, and Hashing for researchers working with data from various modalities to solve sentiment analysis, emotion recognition, and information retrieval tasks. We introduced the importance of learning joint representation and temporal data to capture inter-modal and intra-modal interactions in the data. We also presented hashing techniques for data representation. They have emerged as a low-cost solution in terms of prediction time and memory. This review provides details about a unified representation learning framework, few deep fusion architectures, a few deep and non-deep hashing methods, commonly used datasets, a comparison between methods, and an overview of MMRetrieval(real-life multimodal retrieval system). Studies suggest that there is a gap between research and practice. There is a tradeoff between good quality results and execution time. As a future direction, it is necessary to design a good quality scalable multimodal retrieval system.

CCS Concepts: • **Multimodal Machine Learning**; • **Information Retrieval**;

Additional Key Words and Phrases: multimodal representation, multimodal fusion, hashing, datasets

ACM Reference Format:
Shivangi Bithel, Dwijesh Gohil, and Prabhat Kanaujia. 2021. A Selective Survey on Multimodal Representation, Fusion and Retrieval . *ACM Trans. Graph.* 37, 4, Article 111 (August 2021), 14 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Modality refers to how something occurs or is experienced, and a research problem is defined as multimodal when it includes multiple modalities like image, audio, and text.

This review starts with an introduction to representation learning for multimodal data, which is about learning joint representations across multiple modalities.This section on representation learning sets the background for the subsequent topics covered in this survey, starting with the need of learning joint representations and proceeding up to talking about a unified representation learning framework that spans across multiple datasets and tasks.

Building up on Section 2, Section 3 has defined various deep multimodal fusion architectures that fuse different intermediate representations of audio, text, and image data. There are three types of fusion defined in the literature, Early fusion, Late fusion, and Hybrid fusion. We are trying to capture the data's inter-modal and intra-modal interactions by fusing these different representations. Having data available in multiple modalities provides robustness to the model, and we can learn richer vector representations. If any modality data is missing or noisy, other modality data can help in providing a representation. Architectures like MFN[Zadeh et al. 2018], TFN[Zadeh et al. 2017], LMF[Liu et al. 2018], T2FN[Liu et al. 2018], Deep-HOseq[Verma et al. 2020], Auto-Fusion and GAN-Fusion[Sahu and Vechtomova 2021] use initial vector representation for audio, visual, and language data. We combine these vectors/tensors to form a single common representation that captures view-specific and cross-view-specific interactions of the data and learns the meaning of the data similar to the humans. We have also defined an autoML model called MFAS, which searches for the efficient architecture to fuse the different modality data. It starts from layer one and increases the model's complexity to find the most efficient architecture for fusion. Following Fusion, we also talk about hashing methods and their relevance in retrieval in multimodal setting, in Section 4. Hashing methods are broadly divided into two categories: deep methods and non-deep methods. Specifically, SDML[Hu et al. 2019], DSMHN[Li et al. 2019], SRLCH[Shen et al. 2020], and MFMH[Zeng et al. 2019] methods have been reviewed in detail. SDML, DSMHN and MFMH are deep methods and SRLCH is non-deep method. All four methods are compared under four factors: training time, adding new modalities, integer constraint relaxation, and dataset usage. In multimodal retrieval field, few datasets are widely used: NUS-WIDE, MIRFlicr25k, MS-COCO, Wiki, Lableme, etc. Section 5 contains comprehensive details about NUS-WIDE, MS-COCO and Wiki dataset. Wherever possible, information on dataset collection procedure, ground truth categories, features, and train-test-val split is provided. Having so many state of the art method is good, but it is interesting to see the gap between research and practice. MMRetrieval is one such real life multimodal retrieval search engine. Which after investigation shows that the real life system still needs to answer scalability and quality aspects. In Section 6, we provide the comparison of different practices described in this survey. In Section 7, we reviewed a real-life multimodal search engine. We concluded the survey with some open-ended questions for future research in Section 8.

2 REPRESENTATION LEARNING

Multimodal representation of information from different sources can be much more informative than any of those sources separately. This section aims to provide an insight into the feasibility, challenges and directions to work on, for learning joint representation for different modalities. Since there is quite a lot of variance in the

*All authors contributed equally to this research.

Authors' address: Shivangi Bithel, csy207657@cse.iitd.ac.in; Dwijesh Gohil, cs5170407@cse.iitd.ac.in; Prabhat Kanaujia, cs5160789@cse.iitd.ac.in, Indian Institute of Technology, New Delhi, India.

0730-0301/2021/8-ART111 $15.00
https://doi.org/10.1145/1122445.1122456

properties of the data from different modalities, the task of joint representation learning is not a trivial one. We start our survey with a paper [Collell and Moens 2018] which shows that moving between different representations is not as simple as learning a mapping from one modality to another. This paper shows that the semantic structure is very hard to transfer between modalities by a simple mapping. Going forward, research has been conducted into learning joint representations for different modalities [Yang et al. 2017], [Wu et al. 2019] and the generalization of representation learning over various datasets and tasks [Lu et al. 2020]. These give a sense into the general direction in which representation learning seems to be moving and an insight into the further improvements required.

2.1 Bridging Modalities using Cross-Modal Mappings [Collell and Moens 2018]

This paper serves as the starting point for our survey as it studies the feasibility of bridging the representation of different modalities using neural networks and analyses the same, quantitatively, via novel evaluation metric.
More specifically, the paper:

(1) takes embeddings in the text domain(GloVe [Pennington et al. 2014], word2vec [Mikolov et al. 2013], bidirectional GRU features) and the image domain(last layer of VGG-128 [Chatfield et al. 2014], ResNet [He et al. 2015]) on ImageNet [Russakovsky et al. 2015], IAPR-TC-12 [Grubinger et al. 2006] and Wiki.
(2) defines linear and neural architectures(trained and untrained, both) to serve as a map(represented by function **f(.)** from one domain(X) to the other(Y)
(3) Compares the neighbourhood structure of the predicted vector f(X) to X and Y, using mNNO which implicitly quantifies semantic similarity between two sets of paired vectors

$$mNNO^K(V, Z) = \frac{1}{KN} \sum_{i=1}^{N} NNO^K(v_i, z_i)$$

where, $NNO^K(v_i, z_i)$ denotes the number of K nearest neighbours that v_i and z_i share in their respective spaces

Ideally, the mapping should be such that the neighbourhood structure of the predicted vectors(embeddings) is similar to that of the target vectors. The results, however, suggest the following:

- cross-modal mapping produced via training neural models semantically resemble the input vectors more than the target vectors. This shows that directly training a neural network to map one set of existing embeddings to another set might not be very promising and we need to look more into learning joint/fused representations
- Another interesting observation is that even if we take untrained neural networks to map one embedding to another, the resultant outputs are semantically similar to the inputs, according to the $mNNO^K$ metric

Hence, the results of this paper lay the groundwork and the rationale for learning joint embeddings across modalities and to explore multimodal fusion techniques.

2.2 Multimodal Representation Learning from Temporal Data [Yang et al. 2017]

Temporal data offers a dimension around which data from various sources can be structured. This paper considers video and motion-sensor-data modalities and video and audio modalities, which takes advantage of the temporal structure during the embeddings fusion process. The proposed model(**corrRNN**) learns a joint representation between the two modalities. Carefully designed loss terms and ability to dynamically adjust the weightage of each modality (to emphasise more useful signals) in the joint representation also provide valuable insights into the nature of multimodal representations. Additionally, and also most importantly, the proposed approach is easily extendable to include more modalities, apart from the video-sensor(ISI Dataset [Kumar et al. 2015]) and video-audio(AVLetters [Matthews et al. 2002], CUAVE [Patterson et al. 2002]) domains considered in this work.
An interesting point to note here is that the temporal data is significantly different from the other occurrences of multimodal data, in a sense that temporal multimodal data consists of different modalities (representations) of the same event, synced together by time, whereas other forms of multimodal data are usually related more in a semantic aspect only.
Salient features of this work are:

(1) Comprises of an multimodal encoder and decoder that maps input sequences to a joint representation and reconstructs the input sequence from this representation, respectively
(2) Dynamic Weighting module to emphasise on the modality with the more useful signal at a given time step
(3) Loss functions to represent reconstruction losses from joint representation, self representation and cross representation

For each of the pair of modalities, the experiment is designed as a supervised classification task. The proposed temporal model outperforms non-temporal models on video-audio based classification tasks. Key takeaways being:

(1) corrRNN outperforms non-temporal (MDAE [Ngiam et al. 2011], MDBM [Srivastava and Salakhutdinov 2012]) and temporal models (CRBM, RTMRBM[Amer et al. 2018]) in fused video-audio representation setting as well as in cross-modality and shared representation learning setting.
(2) Subsequent inclusion of each of the 3 losses and the dynamic weighting component progressively increases the accuracy of the model on video-sensor(ISI) [Kumar et al. 2015] dataset

Note: Metric used in this paper is the classification accuracy
This addresses some of the drawbacks of multimodal mapping pointed out in the earlier paper, and instead provides a way to learn more robust fused-embeddings that have better cross-modality, via a careful choice of loss functions and a dynamic weighting scheme to choose between different modalities at different timestamps.

2.3 Unified Vision and Language Embeddings [Wu et al. 2019]

The proposed work, Unified VSE(Visual Semantic Embeddings), creates a joint embedding for visual and textual modalities. It unifies concepts at different levels, which, quoting the paper are as follows

- "objects(noun phrases vs visual objects), attributes(pronominal phrases vs visual attributes), relations(verbs/ preposition vs visual relation) and scenes(sentence vs image)."
The authors identify the following challenges:

(1) There are multiple objects in most visual scenes. However, the description, caption, associated text or some other textual representation does not talk about all of these objects. Hence, finding the correct association between these two modalities is often ambiguous
(2) Bias in the dataset (such as almost universal co-occurrence of 2 objects throughout the dataset) can lead to encoders to learn embeddings based on only part of the sentence, which can expose the model to significant adversarial attacks

And the proposed work provides solutions to address these challenges, which can be extrapolated to other tasks from same or different sets of modalities. This paper uses MS-COCO [Lin et al. 2015]. Syntactic dependency parsing extracts object entities, adjectives and prepositional phrases from image captions, thus effectively factorizing the semantic space into various levels. These are then encoded using an unified object encoder ϕ for encoding objects and attributes, and a neural combiner ψ for encoding relations and sentences.

(1) The above parsing, textual encoding and factorization regime allows to establish fine-grained relations between visual and textual data
(2) It also allows for the coverage of all semantic components of the textual input. This leads to the model being defended against textual adversarial attacks by learning a robust semantic representation
(3) The factorized semantic space also allows for contrastive negative sampling which augments direct sampling of negative captions by randomly substituting individual components of attribute-noun pairs or relational triplets as well.
(4) This allows for a "relevance-weighted alignment mechanism" to align specific local regions of image to textual objects, in addition to the global-level image alignment used in previous works

Based on the above improvements, Unified VSE outperforms(**R@k**) previous works on both Image-to-Sentence Retrieval and Sentence-to-Image Retrieval such as HM-LSTM [Niu et al. 2017], multimodal-CNN [Ma et al. 2015] and CSE [Xiao et al. 2017]. It is also more robust to textual adversarial attacks, as compared to VSE++ [Faghri et al. 2018] and VSE-C. [Shi et al. 2018]
This paper, hence, provides a very useful approach of levelized semantic factorization of textual data to learn robust joint representation for vision and language. It also enables contrastive sampling, aligning text to specific local regions in image and enforce semantic coverage of entire caption. All of these can be extrapolated to varying extent in future works and show significant promise.

2.4 Multi-task Representation Learning [Lu et al. 2020]

This paper addresses the lack of generalization in previous works, with regards to the tasks that joint Visual and Language embeddings are targeted at. They develop a model that is trained on 12 different datasets spanning across multiple categories. These include visual QA, MM Verification, Image Retrieval using captions, among others. Key benefits include 90 % reduction in total number of parameters, average performance increase of 2% and further performance improvements via fine-tuning task specific models using the unified model.

The proposal is based on the ViLBERT [Lu et al. 2019] model. It makes certain modifications to the pretraining step of ViLBERT [Lu et al. 2019], which acts as the 'trunk' from which task-specific 'heads' branch off, to facilitate multi-task training, with the goal of learning shared trunk parameters Θ_S that minimize the loss across all 'heads'.
There is also a need for the input to be augmented with task-tokens, to allow the network to follow the correct track. Also, 4 loss functions, one for each track, are defined and have to be minimized for a shared Θ_S. Certain modifications are also needed to handle the varied task-heads and datasets(12) of varying size. The primary goal of these is to handle larger-scale multitask training and specific details can be found in the paper.

To establish a baseline, single-task models are trained on top of the ViLBERT architecture, for each of the 12 datasets. Evaluation is performed as follows:

(1) Intra-group multitask performance - Here tasks within the same group are jointly trained and this model is used for evaluation. 11 out of 12 tasks show improvement in performance
(2) Inter-group multitask performance - Model is trained using a pair of tasks, each representative of its group. Here also, except group 4 (NLVR [Suhr et al. 2019]), other pairwise inter-group trainings yield some performance improvement
(3) Primary contribution is a single model trained on all 12 datasets. This model outperforms 11 out of 12 single-task baselines trained on the same dataset, providing an average improvement of 2.05%, while at the same time reducing the number of parameter by more than 90 %.

Overall, this work provides a framework for multi-modal learning, generalized to combine multiple tasks on a large-scale, which offers significant gains in terms of performance and efficiency and can also serve as a pretraining step over which better task-specific models can be obtained via fine-tuning. It also provides a generalized multimodal representation, that is defined primarily by the modalities under consideration and can be extended to include multiple tasks/datasets.

3 MULTIMODAL FUSION

Multimodal fusion is a fundamental method of "multimodal data mining which aims to combine the data from different sources, distributions and types into a global space" in which both view-specific and cross-view specific interactions can be represented in a consistent manner. It can provide more valuable information than an individual modality by leveraging modality-specific information.[Gao et al. 2020] Motivations for data fusion include obtaining a more associated picture and global view of the system, improving decision making by providing robust predictions, exploratory research, leveraging the complementarity of homogeneous data, ability to provide predictions if one of the modalities is missing.

The difficulty arises due to the different granularity of the involved modalities. [Atrey et al. 2010]
Multimodal fusion has a broad range of applications like audio-visual speech recognition,text-to-speech conversion, speech-to-text-conversion, sentiment analysis, question answering system, graphic matching, text description, emotion recognition, medical image analysis, multimedia description, or information retrieval. Studies have shown that more efficient fusion methods translate to better performance in models, and there has been a wide range of fusion methods. [Baltrušaitis et al. 2017]

3.1 Types of Fusion Methods:

(1) Early fusion is a method that uses feature concatenation to fuse data from different modalities. It is easy to implement and exploit dependencies between features. The drawback here is that there can be a loss in intra-modal interactions because of fusion at an early stage. Also, these features can end up having high dimensionalities. Early fusion approaches give classification output and cannot capture temporal features. They are primarily used in emotion recognition tasks.
(2) Late Fusion is a technique that builds separate models for each modality and then integrates the output using a method such as a majority voting, weighted sum, or a machine learning approach. The disadvantage of this class of methods is that it requires multiple training stages to create separate models for each modality. It is not efficient in capturing low-level interactions or inter-modal interactions. Late fusion approaches give regression output and can be used to capture temporal features. They are primarily used in emotion recognition tasks.
(3) Hybrid Fusion approaches tend to take advantage of both early, and late fusion approaches explained above. The fusion can be done using any intermediate representation by passing it through a non linearity function such as ReLU to give a joint representation of the multimodal data. It gives classification output and are primarily used in multimedia event detection tasks.

3.2 Types of interactions we capture in multimodal representations:

(1) View specific or Intra-modal interactions: These interactions involve only one view(modality), like learning the speaker's sentiment based only on the sequence of spoken words or written text.
(2) Cross-View specific or Inter-modal interactions: These are the interactions defined across different views. They span across both views and time. For example, learning the speaker's sentiments in a video based on the sequence of spoken words, facial expressions, and pitch of the sound.

3.3 Multimodal Fusion Architectures:

This section provides an explanation to the deep multimodal fusion architectures like MFN[Zadeh et al. 2018],TFN[Zadeh et al. 2017], LMF[Liu et al. 2018], T2FN[Liu et al. 2018], Deep-HOseq[Verma et al. 2020], Auto-Fusion and GAN-Fusion[Sahu and Vechtomova 2021], SDML[Hu et al. 2019] and MFAS[Pérez-Rúa et al. 2019].

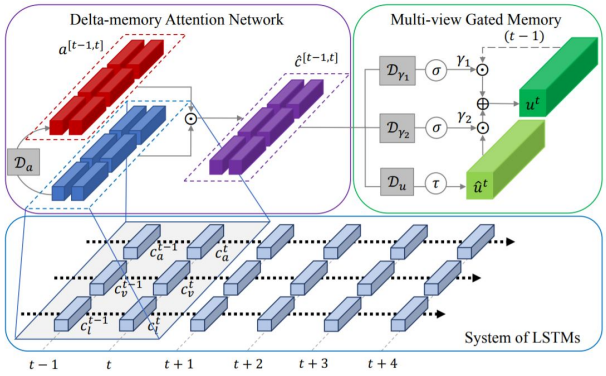

Fig. 1. Memory Fusion Network pipeline.

3.3.1 Memory Fusion Network for Multi-view Sequential Learning [Zadeh et al. 2018]. Memory Fusion Network is a neural model for multi-view sequential learning. MFN accounts for view-specific and cross-view-specific interactions and continuously models them through time with a unique attention mechanism and investigated through time with a multi-view Gated memory. It contains three layers, as can be seen in Figure 1. The first layer is called the System of LSTMs. It consists of multiple Long-Short Term Memory networks, one for every view. They are used to encode view-specific interactions over time. The problem that this temporal layer solves is the alignment. If the cell from modality three at time t is related to the cell from modality one at time t-1, how can we fuse them? This layer takes local pieces of evidence, performs local fusion, and then updates a memory based on the output. The second layer is Delta-memory Attention Network(DMAN). It uses an attention mechanism to capture cross-view and temporal interactions. It takes into input the concatenation of two memories at t-1 and t and compares them by passing them through a neural network and obtaining the attention coefficients. It assigns high coefficient values if the state of memories in the System of LSTMs changes. It applies softmax activated score at the neural network's output layer and allows regularizing the high-value coefficients over concatenated memory. It helps preserve the dimensions of the input memory vector, which are highly important, and neglects the ones that are not important. The third layer is Multi-view Gated Memory which stores the cross-view interactions. It is a unifying memory for the memories in the System of LSTMs. It is controlled using a set of two gates called retain and update gates. Retain gate assigns, how much of current state to remember at time t and update gate assigns, how much of the memory to update. The MFN outputs are the final state of the Multi-view Gated Memory and the results of each of the n LSTMs. MFN was able to achieve state-of-the-art performance for the multi-view sequential modeling task.

3.3.2 Efficient Low-rank Multimodal Fusion with Modality-Specific Factors [Liu et al. 2018]. In this paper, the author proposes a Low-rank Multimodal Fusion method that is efficient in computing tensor-based multimodal representations with fewer parameters and computational complexity. It can scale linearly in the number of modalities. [Zadeh et al. 2017] introduces Tensor fusion network, which models the view-specific and cross-view dynamics. It creates a multi-dimensional tensor that captures unimodal, bimodal, and trimodal interactions across three modalities. It computes the outer product between unimodal representations from three different modalities to compute tensors. It is a successful approach but requires transforming input representation to high dimensional tensor and then mapping it back to lower-dimensional vector space. Also, the dimensionality of tensor increases exponentially with the number of modalities, in this case, making it computationally inefficient with a large number of modalities. LMF (Low-rank multimodal fusion) makes the task of generating these multimodal representations in linear time with the number of modalities by using the naturally decomposable tensor Z and modality-specific low-rank factors to decompose the weight tensor W using canonical polyadic decomposition. The minimal *r* for tensor decomposition is called the rank of the tensor. This method decomposes W into factors equal to the number of modalities, thus modality-specific decomposition. This approach combines the modality-specific characteristics from each of the modalities and generates modality-specific intermediate representation. We get a final multimodal representation h, as shown in the Figure 3, on multiplying these intermediate representations. The authors of this paper made no comments regarding the presence of noise in the data and missing data, which can lead to higher rank tensor representation.

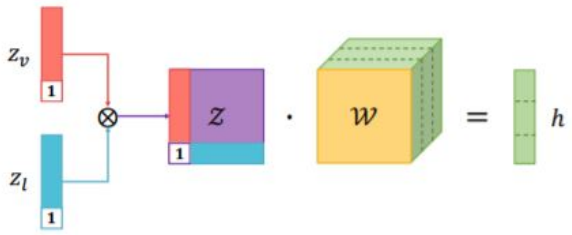

Fig. 2. Tensor fusion via tensor outer product.

3.3.3 Learning Representations from Imperfect Time Series Data via Tensor Rank Regularization [Liang et al. 2019]. This paper solves the problem of the presence of noise and the missing data, which breaks the correlation present in time series data and leads to higher rank tensor representation which LMF cannot solve. This paper introduces a new model, T2FN (Temporal Tensor Fusion Network), to build tensor representations for time series data, using tensor rank minimization as a regularizer for training noisy data. In the Figure 4, the LSTM network is used to encode the temporal information from each modality. Tensor *M* is a multimodal representation created using outer products of individual representations of temporal data. The rank of M increases with noisy and missing data. Tensor rank

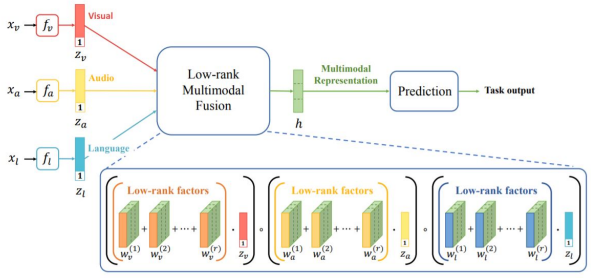

Fig. 3. Low-rank Multimodal Fusion model structure

regularization is used to provide a minimum upper bound on the rank of M. The paper experiments the effect of noise levels on the value of rank. M can be used to find similarity metrics for ranking and retrieval tasks.

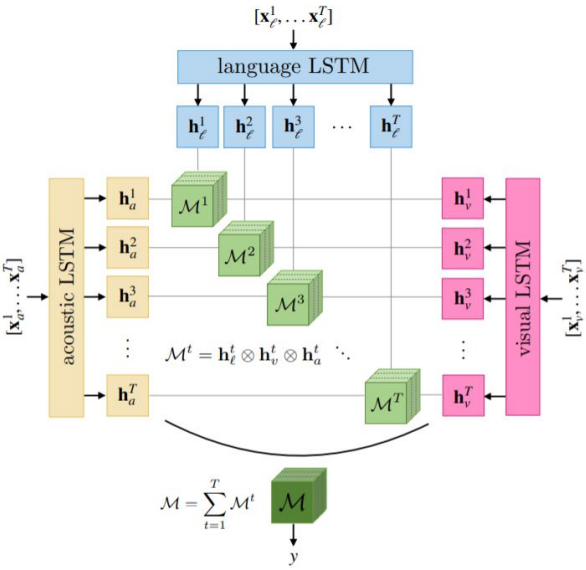

Fig. 4. Temporal Tensor Fusion Network structure

3.3.4 Deep-HOSeq: Deep Higher Order Sequence Fusion for Multimodal Sentiment Analysis [Verma et al. 2020]. The Deep-HOseq extracts two kinds of information from multimodal time-series data. First is an amalgamation of view-specific and cross-view specific information extracted from a common sub-network in a cascade manner. Second is the unique sub-network, which extracts the temporal-granularity within the modalities. Information extracted from both unique and common sub-network is then integrated using a fusion layer, as shown in the Figure 5. The common sub-network first processes the unidirectional LSTMs to extract intra-modal dynamics. The obtained latent features are then processed through a fully connected layer succeeded by an outer product to get multimode tensors.

$$h_v = \sigma(LSTM(z_v) * W_v + b_v)$$

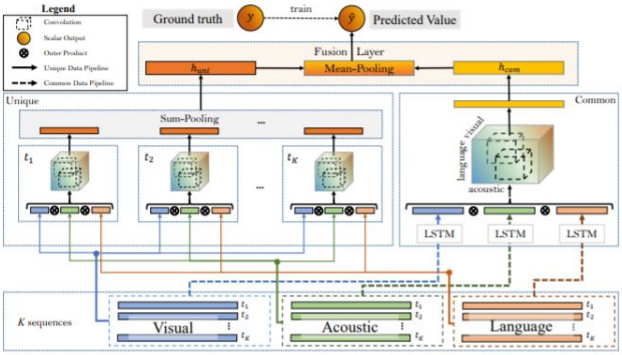

Fig. 5. Deep-HOseq structure

$$h_A = \sigma(LSTM(z_a) * W_a + b_a)$$
$$h_L = \sigma(LSTM(z_l) * W_l + b_l)$$
$$T_{VAL} = h_v \otimes h_A \otimes h_L$$
$$G_{VAL} = \sigma(Conv(T_{VAL}))$$
$$h_{com} = \sigma(h_n \times W_{(com)} + b_{com})$$

h_{com} denotes the amalgamated view-specific and cross-view specific information. The unique network uses feed-forward layer to derive and intermediate representation which it further passes through convolution and fully connected layers to extract cross-categorical correlations. Here k = 1,2,...t_k

$$h_{V_k} = \sigma(z_{vk} \times W_{vk} + b_{vk})$$
$$h_{A_k} = \sigma(z_{ak} \times W_{ak} + b_{ak})$$
$$h_{L_k} = \sigma(z_{lk} \times W_{lk} + b_{lk})$$
$$T_{VAL_k} = h_{V_k} \otimes h_{A_k} \otimes h_{L_k}$$
$$h_k = \sigma(\sigma(Conv(T_{VAL_k})) \times W_{val_k} + b_{val_k})$$
$$h_{pool} = \sum_{k=1}^{t_k} h_k$$
$$h_{uni} = \sigma(h_{pool} \times W_{pool} + b_{pool})$$
$$h_{combined} = average_pooling(h_{com}, h_{uni})$$

$h_{combined}$ denotes the combined multimodal tensor, which can be further used to find similarity metric for retrieval task.

3.3.5 Adaptive Fusion Techniques for Multimodal Data [Sahu and Vechtomova 2021]. In this paper, the author proposes two architectures for fusion of multimodal data. First is **Auto-fusion**, a fusion technique that learns to compress information from different modalities while preserving the contextual meaning. It extracts the multimodal features by maximizing the correlation between input modalities. As observable from the Figure 6, we first concatenate each unimodal feature z_{m1}^{d1}, z_{m2}^{d2} and z_{m3}^{d3} to obtain z_m^k. It is then passed through a transformation layer T, reducing its dimension to t, to obtain autofused latent vector z_m^t. We reconstruct the original vectors by passing z_m^t through another transformation layer F_c to obtain $\hat{z}_m^k$. The loss function is calculated as the Euclidean distance between original and reconstructed vectors. Intermediate vector $\hat{z}_m^t$ is considered as the fused multimodal representation.

$$J_{tr} = \left\| \hat{z}_m^k - z_m^k \right\|^2$$

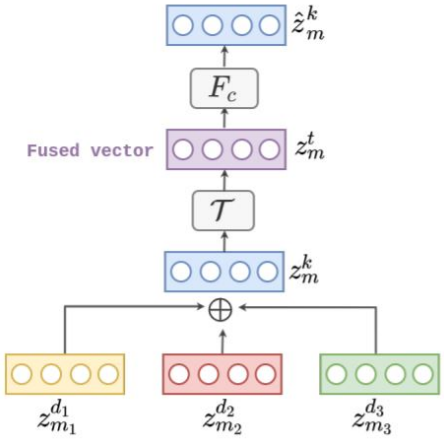

Fig. 6. Auto-Fusion Network

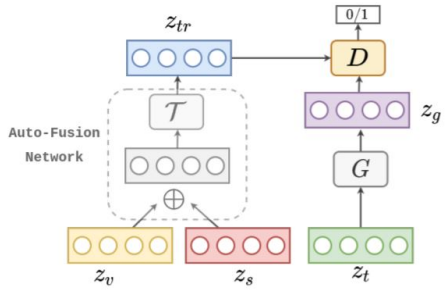

Fig. 7. GAN-Fusion Network

The second method is **GAN-fusion**. The architecture of GAN based fusion model can be observed in the Figure 7. It employs an adversarial network that regularizes the learned hidden space for a given target modality complying with the learning offered by complementary modalities. It helps in disambiguation of challenging sentences. First every modality is represented in their respective latent vectors z_s, z_v and z_t, and one of them is chosen as a target modality. The other two are fused and passed through a transformer layer to reduce dimension and represented as z_{tr}. The target modality is passed through a generator layer G to form z_g. The network will be trained in adversarial fashion with labelling z_g as negative samples and z_{tr} as positive samples. The loss function for a single modality is given as:

$$\min_G \max_D J_{adv}^t(D,G) = \mathbb{E}_{x \sim pz_{tr}(x)}[logD(x)] + \mathbb{E}_{z \sim pz_t(x)}[log(1 - D(z_g))]$$

Total adversarial loss is given by:

$$J_{adv} = J^t_{adv} + J^s_{adv} + J^v_{adv}$$

z_{fuse} can now be used for various tasks like generation, classification, ranking and retrieval tasks.

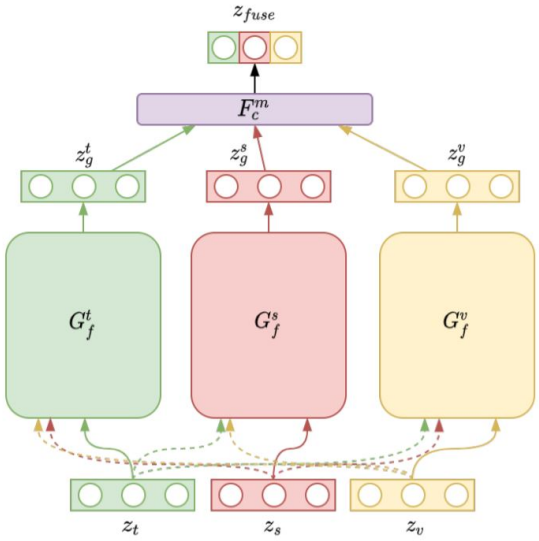

Fig. 8. Overall architecture of the GAN-Fusion module.

3.3.6 Scalable Deep Multimodal Learning for Cross-Modal Retrieval [Hu et al. 2019]. SDML uses DSAE (deep supervised auto-encoder) to transform the samples from each modality into a common subspace. It is an auto-encoder with an extension of a supervised loss on the representation layer, inferred from the label information to push possible discrimination into the predefined common subspace. In the Figure 9 we can observe an encoder. a decoder and a supervised label projection. Encoder is denoted as $h^i_j = f_i(x^i_j, \Theta_i)$ and it projects the modality data into a predefined common subspace. Decoder is denoted as $\hat{x}^i_j = g_i(h^i_j, \Phi_i$ for the i-th network and reconstructs the sample from the common subspace representation. We get the following objective function for DSAE:

$$\tau^i = \frac{1}{n_i}\sum_{j=1}^{n_i}[\lambda\tau^i_r(x^i_j) + (1-\lambda)\tau^i_s(x^i_j)]$$

The loss term contains λ as a balanced parameter to trade off between the supervised loss $\tau^i_s(x^i_j$ and reconstruction error $\tau^i_r(x^i_j)$. P is a fixed matrix which projects the sample representations from predefined common subspace to label space. The objective function for SDML is to simultaneously minimize the above equation for m modality specific networks. SDML is scalable as each modality has its independent parameters to learn and that can be done parallely. For testing, the encoder's outputs are used as common representations of the samples. One can find similarity metric like cosine similarity for ranking and retrieval task.

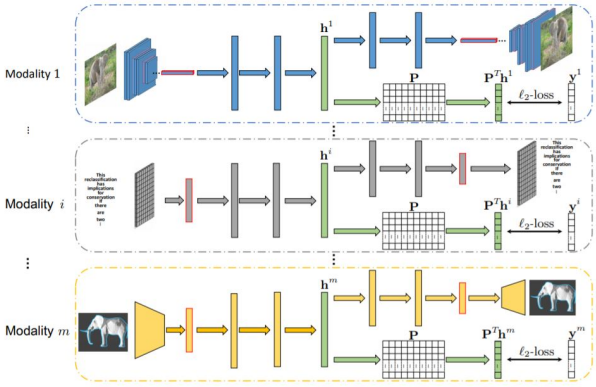

Fig. 9. SDML method Framework

3.3.7 MFAS: Multimodal Fusion Architecture Search [Pérez-Rúa et al. 2019]. The paper focuses on finding a hybrid fusion architecture to learn the multimodal data's joint representation in an AutoML way. MFAS explore the search space with sequential model-based optimization, starting with simpler models for L=1 and iteratively increasing the complexity for L=2, L=3, etc. It uses a surrogate function to predict the performance of unseen architectures. The knowledge learned in L=1 can be used to derive a subset of plausible architectures instead of trying every combination in L=2, thus reducing the computation. Similarly, subsequent layers will use the knowledge gained from all previous layers. A single fusion unit for the Bi-modal fusion network takes layer index from modality x and modality y having M and N layers, respectively, and an activation or fusion function as a hyperparameter to be learned, as can be seen in the Figure 10. The paper made no comment on the individual representations of the modalities. They are assumed to be pre-trained on an efficient model. One can develop an new method which not only finds a model for fusion but also finds an efficient way to extract features from individual modalities. There are other Neural architecture search methods like DenseNAS [Fang et al. 2020], RandomNet [Alletto et al. 2020], etc developed after this approach. RandomNet proposes new evaluation methods for multimodal NAS like measuring the degree of human intervention. MFAS has been experimented with Audio-visual MNIST dataest, multimodal IMDB dataset and NTU RGB+D dataset for finding efficient architectures that can solve dataset specific tasks effectively.

4 HASHING TECHNIQUES

Hashing methods are broadly divided into two categories: deep methods and non-deep methods. SDML[Hu et al. 2019], DSMHN[Li et al. 2019], and MFMH[Zeng et al. 2019] are deep methods and SRLCH[Shen et al. 2020] is non-deep method. Underlying principles for all four methods are described below.

4.1 Deep Hashing

4.1.1 Deep Semantic Multimodal Hashing Network for Scalable Multimedia Retrieval[Li et al. 2019]. The proposed hashing technique

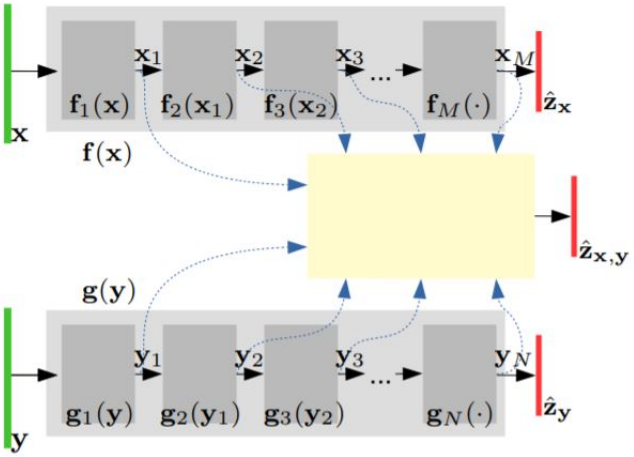

Fig. 10. Bi-modal Fusion network structure.

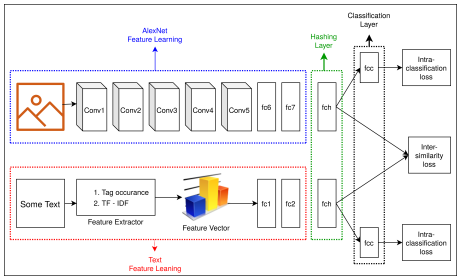

Fig. 11. Deep Semantic Multimodal Hashing Network(DSMSN)

involves a supervised cross-modal CNN(Convolutional Neural Network) based architecture. Figure-11 represents the end-to-end architecture. At high level, it can be divided into 4 parts. **AlexNet** architecture is used to extract features from an input image. **Text feature leaning** architecture uses traditional text features(tag occurrences, TF-IDF) to feed in to the convolutional layers. **Hashing layer** outputs bit vector for different modalities. It uses $sign(x)$ function that returns -1 or 1 depending upon the sign of x. **Classification layer** ensures that the learnt hash codes preserves intra-similarity. It classifies a bit vector into one of the training labels.

The proposed architecture accounts for both "intra" and "inter" similarity between the items. To reason out the claim, consider the loss function description below:

$$\min_{\phi_X,\phi_Y} \Omega = \Sigma\, l_{ij} + \alpha(L_c^X + L_c^Y) + \beta L_q + \gamma L_b$$

ϕ_X, ϕ_Y are the parameters of the network for the image and text modalities respectively. i, j represents items from different modalities. l_{ij} accounts for the inter modality similarity. It measures the difference between the similarity value of item i and item j in original space and in the transformed space. α, β, γ are hyper parameters. L_c^* accounts for the intra modality similarity. Training data has labels associated with each of the text and the images. This loss makes sure that the items after the transformation to the new space(bit vectors) are classified to the correct labels. L_q is a quantization loss which makes sure that the output of each neuron in the **hashing layer** is close to -1 or 1. L_b is a bit balance penalty term. It makes sure that output of each neuron in the **hashing layer** is zero mean. Ideally, the probability of seeing -1 or 1 in the bit vector should be 50%.

One drawback I felt about this method is that it uses CNN for images which makes sense since CNNs are proved to be effective to learn hidden features of the images. But for text also they use CNN. There exists LSTMs, Transformer based networks which can well learn the text features better compared to the CNNs.

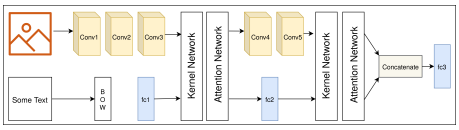

Fig. 12. Architecture for Modal-aware Feature Learning

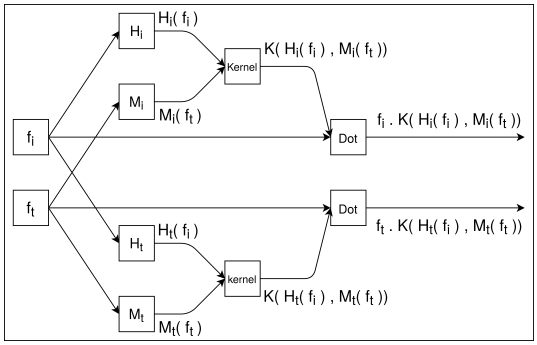

Fig. 13. Kernel network to re-weight features

4.1.2 Modal-aware Features for Multimodal Hashing[Zeng et al. 2019]. This paper[Zeng et al. 2019] proposes a framework to learn better feature vectors for different modalities. Figure-12 represents the end-to-end architecture. At first, the image is passed through three convolutional layers and the text(bag of words) through a fully connected layer to get corresponding intermediate feature vectors. These feature vectors are passed to the kernel network, followed by an attention network. As shown in the figure, the process is repeated to get the final feature vectors for the image and text. As a fusion step, they apply concatenation of feature vectors and pass to fully connected layer to generate the final hash code.

Figure-13 represents the kernel Network. Using such network ensures that the learnt features for one modality knows information about other modalities too. The kernel network takes two feature vectors as an input: f_i(for image modality) and f_t(for text modality). For a given modality, say for an image modality, it applies the following operation: $f_i' = f_i \;.\; K(H_i(f_i),\; M_i(f_t))$. Here H_i and M_i are convolutional layers that maps the input feature vectors to a new space. The authors use $K()$ function as a matrix multiplication. Overall, in this step, the $K()$ output is used to re-weight the original input feature vector.

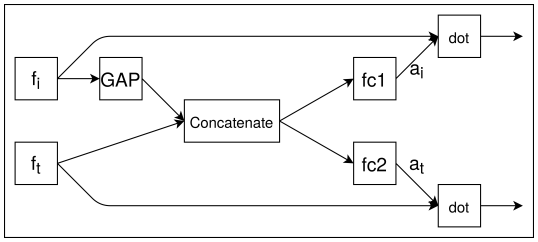

Fig. 14. Attention network to get informative regions

Figure-14 represents the attention network. It is used to figure out the informative regions of the input feature vectors. In Figure-14, GAP stands for Global Average Pooling. In this network, image feature vector is passed through GAP(to align the dimension with that of the text feature vector). Two feature vectors are concatenated. The result is passed to two different fully connected layers($fc1, fc2$) as shown. Softmax function in these fc layers to get the output near to 0 or 1. The output of these layers are: a_i and a_t and are also called as attention maps for corresponding input features. Semantically, they represents the informative regions of the input feature vectors. Finally, non-informative features are ignored by taking the dot product of feature vector with corresponding attention maps.

4.2 Non-deep Hashing

4.2.1 Exploiting Subspace Relation in Semantic Labels for Cross-modal Hashing[Shen et al. 2020]. The proposed method is a supervised cross-modal hashing technique that incorporates: inter and intra modal similarity via exploiting the semantic label information, one step encoding hash function, loss function that reduces the hubness issue, close form solution to the loss function, and an efficient algorithm for training and retrieval.

Hashing Function: Consider $f_m^i \in R^p$ be the p dimensional feature vector for i^{th} item from modality m. To project the f_m^i into a common hamming space, the authors apply non-linear transformation(known as the kernel trick) using $\phi()$ function, followed by a linear transformation using the P_m transformation matrix:

$$h(f_m^i) = sign(\phi_m(f_m^i)\ P_m) \qquad where$$

$$\phi(x) = \left[exp(\frac{-||x - a_1||}{2\sigma^2}),\ \dots,\ exp(\frac{-||x - a_m||}{2\sigma^2}) \right]^\top$$

Where $\{a_i\}_1^m$ are randomly chosen anchor samples, σ is a width, and P_m is a transformation matrix. The label matrix $Y = [Y_1, \dots, Y_n]$ is also transformed in the same hamming space using a transformation matrix W as: $Y_{new} = YW$. Consider b_{m1}^i, b_{m2}^i, and b_l^i are the transformed bit vectors for the item i from modality $m1, m2$ and corresponding label l respectively. Then the overall idea is to reduce the distance between (b_l^i and b_{m*}^i). Following **loss function** reflects such idea.

$$\min_{P_{m1},P_{m2},B,W} \Bigg(||B-YW||_F^2 + v_{m1}||B-h(X_{m1})||_F^2 + v_{m2}||B-h(X_{m2})||_F^2 + \lambda||W||_F^2 + \alpha||P_{m1}||_F^2 + \beta||P_{m2}||_F^2 \Bigg)$$

$$s.t.\ \ B \in \{-1, 1\}^{n\ X\ L}$$

Where Last three terms corresponds to the regularization terms and $||.||_F^2$ denotes the Frobenius norm. n represents the number of training instances and L represents the bit vector length. The below **iterative algorithm** makes use of the close form solutions[1] to find local optimal solutions:

`Step-1:` Fix B, W and update P_{m1}, and P_{m2}. `Step-2:` Fix P_{m*}, B and update W. `Step-3:` Fix P_{m*}, W and update B. Follow `Step-1, 2, 3` until loss function described above converges.

The **training time complexity** is $O(nLTd^2)$ where $d = max(m, c)$. Here n is number of training instances, L is bit vector length, T is number of iteration to converge, m is number of anchors used in the kernel trick, c is the number of class labels.

The proposed loss function also tries to reduce the 'hubness' phenomenon. Consider the following loss function:

$$\min_W ||XW - Y||_F^2 + \lambda||W||_F^2$$

'Hubness' occurs when a high dimensional example space(X) is mapped to a low dimensional label space(Y) using a transformation matrix(W). It is found that $||XW||_2 <= ||Y||_2$. W predicts the same labels more frequently called as 'hubs'. To reduce the impact of 'hubness', the proposed loss function maps low dimensional label space(Y) into a high dimensional hamming space(YW).

5 MULTIMODAL DATASETS

In multimodal retrieval field, MS-COCO[Lin et al. 2015], NUS-WIDE[Chua et al. 2009], and wiki are more commonly used datasets while in multimodal fusion, CMU-MOSI[Zadeh et al. 2016], CMU-MOSEI[Bagher Zadeh et al. 2018] and IEMOCAP[Busso et al. 2008] are more commonly used for sentiment analysis and emotion recognition task. These datasets are reviewed as follows.

5.1 Microsoft COCO[Lin et al. 2015]

The underlying objective behind MS-COCO dataset is to alleviate the object detection and segmentation task. MS-COCO(Microsoft Common Objects in COntext) dataset aims to provide non-iconic views of objects and to provide accurate object labeling and corresponding 2d bounding boxes. Unlike iconic views, MS-COCO dataset images include natural scenes with many objects per image such that the given set of objects determines the scene. Objects can be occluded, may not be present in the middle of the image, and objects may not occur solely in an image. MS-COCO tries to capture images as natural as possible. MS-COCO dataset has 328,000 images with 2,500,000 labeled instances and 91 object categories(11 super-categories to speed up the labeling procedure). Categories only include "thing" categories("sofa", "Motorbike", "Train", etc.) not the "stuff" categories("sky", something that does not have fine boundary). The 2015 release contains 165,482 train, 81,208 validation, and 81,434 test images. Figure-15 represents the 91 categories and number of labeled instances per category. 2014 release contains caption information per image. The way one can use MS-COCO dataset in multimodal retrieval is that use image as one modality and use corresponding caption text as another modality. Now each of the image-caption pair also have assigned category information(out

[1]Exact expressions for close form solutions are not included in this report but can be found in the original paper

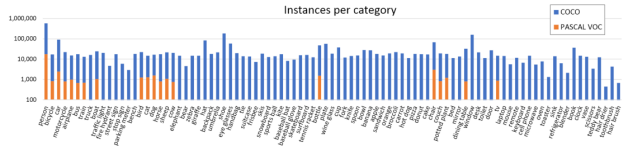

Fig. 15. Number of labeled instances per category vs category for MS-COCO

of 91 categories), which can be used as a ground truth vector. Similar to almost all cross-modal retrieval papers, one can assume that two items of a given modality are similar if they share at least one category.

5.2 Wiki

Wiki dataset[2] contains image, text and corresponding category information. The authors choose a section(text) and an image from a Wikipedia article if the section contains exactly one image and at least 70 words. More precisely, it contains 2173 training images, 693 testing images, 2173 training text, and 693 testing text. There are 10 categories: art, biology, geography, history, literature, media, music, royalty, sport, and warfare. Text is represented as 10D LDA(Latent Dirichlet Allocation) based feature vector. Image is represented as 128D bag of visual SIFT features. Figure-16 shows the

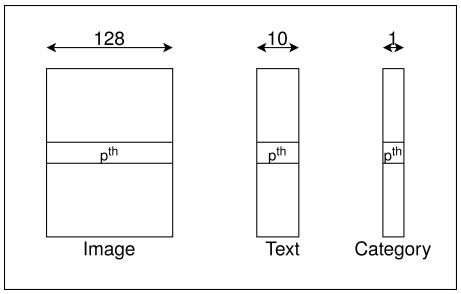

Fig. 16. Data Structure layout of Wiki dataset

data structure layout. p^{th} pair(p^{th} image, p^{th} text) is assigned the p^{th} category($\in \{1, 2, ..., 10\}$).

5.3 NUS-WIDE[Chua et al. 2009]

NUS-WIDE is a multi-label image dataset[3] that comprises of low level features of images, corresponding tags, and ground truth concepts. It has features, tags and ground-truth concepts for 269,648 images. It provides six types of features per image: 500D sift descriptors, 225D block-wise color moments, 128D wavelet texture, 73D edge direction histogram, 144D correlogram, and 64D color histogram. There is one-to-one mapping between image features, 1000D tag vector and 81D ground truth concept vector. 1000D tag vector per image is fetched from flickr website. 81D ground truth concepts are annotated manually for all images in the dataset. Some example concepts are: lake, sand, train, water, etc. In multimodal retrieval field, researchers use image features as image modality, 1000D tags as text modality, and 81D ground truth concepts as corresponding ground truth vectors.

[2]Dataset and related info can be found here: http://www.svcl.ucsd.edu/projects/crossmodal/

[3]Dataset is publicly available to download from: https://lms.comp.nus.edu.sg/wp-content/uploads/2019/research/nuswide/NUS-WIDE.html

5.4 CMU-MOSI [Zadeh et al. 2016]

MOSI is Multimodal Corpus of Sentiment Intensity and Subjectivity Analysis in Online Opinion Videos. This dataset includes opinion videos gathered from YouTube from 93 unique speakers, where each video has multiple opinion segments with a total of 2199 utterances in the complete dataset. The pieces from each video are annotated with sentiments in the range of [-3, 3]. +3 is a positive sentiment, while -3 is a negative sentiment.

5.5 CMU-MOSEI [Bagher Zadeh et al. 2018]

MOSEI is Multimodal Opinion Sentiment and Emotion Intensity. This dataset is the largest dataset for multimodal sentiment analysis and emotion recognition. It includes 23,453 annotated sentences from 1000+ online speakers. It follows the similar annotation from CMU-MOSI dataset i.e. Sentiments lies in the range of [-3, 3], where -3 is a negative sentiment and +3 is a positive sentiment.

5.6 IEMOCAP[Busso et al. 2008]

IEMOCAP is an interactive emotional dyadic motion capture database. This dataset consists of 151 videos of recorded dialogues, with two speakers per session for 302 videos across the dataset. Different video segments are annotated with nine different emotions angry, excited, fear, sad, surprised, frustrated, happy, disappointed, and neutral. Dataset also provides annotation for valence, arousal, and dominance.

6 EXPERIMENTS

This section provides a comparison of different types of methods explained in above sections. We test the performance of these methods based on evaluation metrics and cite the reported results from their corresponding papers in the literature.

6.1 Comparison of Fusion Architectures

The above architectures are tested against classification and regression tasks like sentiment analysis, emotion recognition, speaker trait analysis, etc. using CMU-MOSI, CMU-MOSEI, POM, IEMOCAP, and other datasets.

6.1.1 Sentiment Analysis: The task is to identify the sentiment of the speaker based on online video content. Datasets used for this task are CMU-MOSI and CMU-MOSEI. They use three modality features, namely language, visual and audio. The language model is trained using Pre-trained 300-dim Glove Word Embeddings. COVAREP acoustic framework is used to generate audio features. For extracting Visual features for each frame (sampling = 30Hz), Facet library is used. These feature vectors are publically available. CMU-MOSI uses 1284 training instances, 299 validation instances, and 686 testing instances, while MOSEI uses 15290 training instances, 2991 validation instances, and 4832 testing instances. Table 1 and Table 2 compares the performance of MFN, LMF, TFN and DeepHOseq architectures for regression and multi-class classification (7

sentiments). Mean Absolute Error (MAE) and Pearson's Correlation (Correlation) are calculated for regression and Accuracy for multi-class classification. We can observe that MAE decreases from TFN to LMF and from LMF to MFN from the tables. Deep-HOseq performs best on the MAE metric. Similarly, Correlation increases from TFN to LMF and from LMF to MFN, and Deep-HOseq performs best on Correlation. Accuracy increases from LMF to TFN and from TFN to MFN for both the datasets in the case of multi-class classification. Deep-HOseq performs best in Accuracy on MOSI and MOSEI. T2FN is also evaluated using the MOSI dataset. Compared to TFN, the Accuracy of T2FN with regularization is more than TFN on noisy and missing data.

6.1.2 Emotion Recognition: The task is to identify the emotions of the speaker based on the verbal and non-verbal behavior of the speaker. MFN, LMF, GAN-fusion, and Auto-fusion have been tested using the IEMOCAP dataset for this task. These papers report precision and F1 score for different discrete emotion sets and thus are hard to compare. From the numbers reported in [Liu et al. 2018], we can conclude that LMF performs better than TFN and MFN on [Busso et al. 2008] dataset. From [Verma et al. 2020] we know that GAN-fusion performs better than Auto-fusion, and they are an improvement over LSTMs based baselines.

6.1.3 Other tasks: MFN, TFN, and LMF have been tested for specific trait analysis tasks on POM dataset[Park et al. 2014], and LMF performs better than TFN, and TFN performs better than MFN on MAE, Correlation, and Accuracy. GAN-Fusion and Auto-Fusion are also evaluated on How2 and multi-30K datasets for translation task.

6.1.4 Complexity Analysis: TFN explores the use of tensors for multimodal representation, but increasing modalities in TFN is computationally complex. It uses a cartesian product over unimodal features, which leads to an exponential increase in memory and computation cost on adding new modalities. TFN is a static technique, which means there is no learning procedure involved. LMF is an Adaptive technique, which uses a decomposable module for training and is computationally simpler than TFN. There is no need to convert unimodal representation to the tensors and thus reduces the cost of tensorization. LMF uses 11 times fewer parameters than TFN and is linear in time. T2FN is the extension of TFN, and it also involves the cartesian-product over all involved modalities. The Deep-HOseq method has two networks common and unique. In the common network, we take the outer cross product of vectors from different modalities, which results in a common subspace that is learned jointly in the presence of all modalities. Auto-fusion and GAN-fusion are adaptive techniques that use the individual modalities' concatenation as the initial step and use the concatenated representation for training. Thus all modalities are required for the training as they learn a joint representation.

6.1.5 Use of Temporal Data: Models like MFN, T2FN, Deep-HOseq use the time series data for learning local interactions, while the models like TFN, LMF, Auto-fusion, GAN-fusion do not use temporal data.

Table 1. comparison of Fusion Techniques on CMU-MOSI Dataset [Verma et al. 2020]

Fusion Methods	Regression		7 - class
	MAE	Correlation	Accuracy
MFN	1.0406 ±0.0568	0.5461 ±0.0291	34.14 ±0.0219
TFN	1.1111 ±0.0003	0.5341 ±0.0010	31.98 ±1.1321
LMF	1.0960 ±0.0021	0.5455 ±0.0032	30.76 ±0.0339
Deep-HOseq	1.0201 ±0.0218	0.5676 ±0.0166	35.87 ±0.0332

Table 2. comparison of Fusion Techniques on CMU-MOSEI Dataset [Verma et al. 2020]

Fusion Methods	Regression		7 - class
	MAE	Correlation	Accuracy
MFN	0.7270 ±0.00747	0.5243 ±0.00354	42.69 ±0.00397
TFN	0.7483 ±0.0106	0.5005 ±0.00662	40.88 ±0.0203
LMF	0.7417 ±0.0119	0.5058 ±0.0109	40.64 ±0.00160
Deep-HOseq	0.7189 ±0.00115	0.5438 ±0.00224	44.17 ±0.00260

6.2 Comparison of Hashing Techniques

While discussing the underlying principle for each method, it is also necessary to discuss pros and cons of each methods. Throughout this section, the comparison is carried out for DSMHN([Li et al. 2019]), SRLCH([Shen et al. 2020]), MFMH([Zeng et al. 2019]), and SDML([Hu et al. 2019]) methods with respect to four factors: dataset usage, training time complexity, integer constraint relaxation, and adding new modality.

	Train	Test	Val	# tags
DSMHN	500/tag	100/tag	-	21 most freq.
SRLCH	184,712	1,865	-	not clear
MFMH	10,000	100/tag		21 most freq.
SDML	42,941	23,661	5K	10 most freq.

Table 3. Comparison on NUS-WIDE dataset

	Train	Test	Val	# tags
DSMHN	2,293	573	-	10
SDML	2,173	462	231	10

Table 4. Comparison on Wiki dataset

	Train	Test	Val	# tags
DSMHN	5,000	908	-	24
MFMH	10,000	2,000	-	24

Table 5. Comparison on MIRFlicr25k dataset

6.2.1 Comparing methods on datasets. Table-3, table-4, and table-5 compare DSMHN([Li et al. 2019]), SRLCH([Shen et al. 2020]),

MFMH([Zeng et al. 2019]), and SDML([Hu et al. 2019]) methods (if applicable) for three datasets: NUS-WIDE, Wiki and MIRFlicr25k respectively. NUS-WIDE dataset is used by all four methods but under different settings. Whereas Wiki and MIRFlicr25k are used by two of the methods but again under different settings. Due to the lack of common baseline settings, it is not quite possible to compare the results of the methods.

6.2.2 Comparing on time complexity: Out of all four hashing methods(DSMHN, SRLCH, MFMH, SDML)only SRLCH carries out time complexity analysis experiments, nevertheless we may guess the relative performance of these four methods for training time. SRLCH, being a non-deep method, uses a close form solution to update the weight matrices which can be implemented quite efficiently using a multi-core machine. All remaining methods use traditional neural net learning algorithms for training fairly complex models. With that, it can be inferred that SRLCH should be fastest to train.

6.2.3 Comparing on relaxing integer constraints. In hashing methods, eventually the hash codes has to be discrete to reduce the pediction time. It is quite common to relax integer constraints while training and use quantization loss to minimize information loss. This still gives sub-optimal results. SRLCH doesn't relax the integer constraints. Focus of MFMH is to show that learning better intermediate features by using attention map mechanism leads to higher precision, they do relax the integer constraints in their model. DSMHN also relaxes the integer constraints and introduces quantization loss term to make sure that the information loss is minimal. Whereas SDML entirely ignores the discreteness of hash codes. SDML considers real hash codes for both training and testing. Even the cosine similarity is applied on real hash codes which makes zero information loss at the cost of high prediction time.

6.2.4 Comparing on adding new modalities. Out of all four hashing methods(SDML, DSMHN, MFMH, SRLCH), SDML is specifically designed to incorporate new modalities efficiently without retraining the entire network. SDML uses a common projector matrix P, that projects the output of encoders into the label space. By minimizing the classification loss in that label space preserves the inter and intra similarity. Such design helps them keep neural models for each modality entirely independent of each other. On adding a new modality, SDML needs to only learn a modality specific network by using pre-defined projector matrix P. So SDML can easily incorporate new modalities. Other methods (DSMHN, MFMH, and SRLCH) have to train the entire model from scratch(or use transfer learning) on adding new modalities.

7 MMRETRIEVAL: A REAL-LIFE MULTIMODAL SEARCH ENGINE[Zagoris et al. 2010]

MMRetrieval[4] is an experimental multi-lingual, multi-modal search engine. One can search using image query or text query(in English, France, or German) or both and retrieve images. It provides nice web interface to see the retrieved results. For a given query, it first retrieves results for each modality separately and then merges the results depending upon two main factors: noisiness of modalities and user need(high recall or high precision). It also supports two-stage retrieval. At the core, it uses ImageCLEF 2010 Wikipedia collection[5] database to retrieve results. This collection has 237,434 images and associated noisy annotations. At implementation level, images are represented using compact composite descriptors. Once the results are retrieved for each modalities separately, MMRetrieval can apply any of the fusion technique(user need to select from web interface). Example of one fusion technique: normalize score of each modality results and merge them in sorted order to retrieve. Trying out few queries on web interface, it can be concluded that the system is slow typically on image queries. Sometimes, even a single image query takes 17-18 seconds to retrieve.

8 CONCLUSION AND FUTURE TRENDS

Broadly speaking, multimodal retrieval systems impose two kinds of challenges: retrieving good quality results and designing a scalable system.

Several models in this survey asserted the significance of learning joint representation and the use of temporal data. Starting out, we see in [Collell and Moens 2018] that learning a mapping between two embeddings is not enough. We need a more integrated approach to jointly learn representation for different modalities. [Yang et al. 2017] and [Wu et al. 2019] provide extendable (over modalities) approaches to learn joint representations for temporal and general multimodal data, respectively. And then [Lu et al. 2020] proceeds to provide a unified model for visual and text modalities, that can function across multiple datasets and tasks for these domains, thus providing a more generalizable approach. With possible extensions to include more modalities and more tasks, it seems to provide a path towards an increasingly generalizable model.

Sequence generation models used in tasks like text to speech and speech to text require temporal data to provide an output at every timestamp. The use of LSTMs is prevalent in such models as MFN. Other models like T2FN and Deep-HOseq also provide a way to model sequence data. Different architectures have their pros and cons. Some render more reliable results but are computationally complex, while others are simple but fail on noisy and missing data. Future trends may include modalities like RFID, haptics, etc., for various multimedia analysis tasks. Multilingual data can also be explored as an additional modality to provide a robust representation of the data.

Among the reviewed hashing methods, all of them focus on achieving good quality results and additionally two of them(SDML[Hu et al. 2019], SRLCH[Shen et al. 2020]) focus on scalability aspects as well. Comparison between the methods reveal that there is a tradeoff between good results and execution time. Investigating MMRetrieval indicates the gap between the research and practice

As a future direction, it is suggested that focus should be on designing a good quality scalable multimodal retrieval system. Find ways to fill the homogeneity gap between the original features of different modalities. One way would be to project the features into a common subspace in a way that preserves the inter and intra similarity. Preserving inter and intra similarity is a challenge in its own. It is because projecting high dimensional real data points into

[4]http://mmretrieval.nonrelevant.net/

[5]https://www.imageclef.org/2010/wiki

a low dimensional binary space causes an information loss which usually gives suboptimal results. Current methods also struggle to keep integer constraints in their optimization equation. Usually they relax the constraints and that also results in suboptimal results. Almost all methods come with their own way of filtering the datasets which makes it difficult to compare across various methods. Almost all methods use only two modalities in their evaluation. One reason could be the lack of good quality dataset for other modalities(e.g. audio, video). Second reason could be that unlike images and text, there does not exists state of the art models for other modalities(e.g. Audio, video) that can extract good features from raw data.

Sequence alignment and synchronization can be investigated further as a research problem. AutoML methods like MFAS can also utilize temporal data on layers to generate efficient architectures for sequence generation tasks. Researchers need to explain the dynamics of determined latent space and align multimodal features to address heterogeneity. We require more interpretable features to provide explainability to the black box models.

9 CITATIONS AND BIBLIOGRAPHIES

REFERENCES

Stefano Alletto, Shenyang Huang, Vincent Francois-Lavet, Yohei Nakata, and Guillaume Rabusseau. 2020. RandomNet: Towards Fully Automatic Neural Architecture Design for Multimodal Learning. arXiv:2003.01181 [cs.LG]

Mohamed R. Amer, Timothy Shields, Behjat Siddiquie, Amir Tamrakar, Ajay Divakaran, and Sek Chai. 2018. Deep Multimodal Fusion: A Hybrid Approach. *Int. J. Comput. Vision* 126, 2–4 (April 2018), 440–456. https://doi.org/10.1007/s11263-017-0997-7

P. K. Atrey, M. A. Hossain, Abdulmotaleb El Saddik, and Mohan S. Kankanhalli. 2010. Multimodal fusion for multimedia analysis: a survey.

AmirAli Bagher Zadeh, Paul Pu Liang, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. 2018. Multimodal Language Analysis in the Wild: CMU-MOSEI Dataset and Interpretable Dynamic Fusion Graph. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Melbourne, Australia, 2236–2246. https://doi.org/10.18653/v1/P18-1208

Tadas Baltrušaitis, Chaitanya Ahuja, and Louis-Philippe Morency. 2017. Multimodal Machine Learning: A Survey and Taxonomy. arXiv:1705.09406 [cs.LG]

Carlos Busso, Murtaza Bulut, Chi-Chun Lee, Abe Kazemzadeh, Emily Mower, Samuel Kim, Jeannette N. Chang, Sungbok Lee, and Shrikanth S. Narayanan. 2008. IEMOCAP: interactive emotional dyadic motion capture database. *Language Resources and Evaluation* 42, 4 (Dec. 2008), 335–359.

Ken Chatfield, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. 2014. Return of the Devil in the Details: Delving Deep into Convolutional Nets. arXiv:1405.3531 [cs.CV]

T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. Zheng. 2009. NUS-WIDE: A real-world web image database from National University of Singapore.

Guillem Collell and Marie-Francine Moens. 2018. Do Neural Network Cross-Modal Mappings Really Bridge Modalities? arXiv:1805.07616 [stat.ML]

Fartash Faghri, David J. Fleet, Jamie Ryan Kiros, and Sanja Fidler. 2018. VSE++: Improving Visual-Semantic Embeddings with Hard Negatives. arXiv:1707.05612 [cs.LG]

Jiemin Fang, Yuzhu Sun, Qian Zhang, Yuan Li, Wenyu Liu, and Xinggang Wang. 2020. Densely Connected Search Space for More Flexible Neural Architecture Search. arXiv:1906.09607 [cs.CV]

Jing Gao, Peng Li, Zhikui Chen, and Jianing Zhang. 2020. A Survey on Deep Learning for Multimodal Data Fusion. *Neural Computation* 32, 5 (05 2020), 829–864. https://doi.org/10.1162/neco_a_01273 arXiv:https://direct.mit.edu/neco/article-pdf/32/5/829/1863223/neco_a_01273.pdf

Michael Grubinger, Paul Clough, Henning Müller, and Thomas Deselaers. 2006. The IAPR TC-12 benchmark – a new evaluation resource for visual information systems.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep Residual Learning for Image Recognition. arXiv:1512.03385 [cs.CV]

Peng Hu, Liangli Zhen, Dezhong Peng, and Pei Liu. 2019. Scalable Deep Multimodal Learning for Cross-Modal Retrieval. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Paris, France) *(SIGIR'19)*. Association for Computing Machinery, New York, NY, USA, 635–644. https://doi.org/10.1145/3331184.3331213

Jayant Kumar, Qun Li, Survi Kyal, Edgar A. Bernal, and Raja Bala. 2015. On-the-fly hand detection training with application in egocentric action recognition.. In *CVPR Workshops*. IEEE Computer Society, 18–27. http://dblp.uni-trier.de/db/conf/cvpr/cvprw2015.html#KumarLKBB15

Zechao Li, Lu Jin, and Jinhui Tang. 2019. Deep Semantic Multimodal Hashing Network for Scalable Multimedia Retrieval. *arXiv* (2019). arXiv:1901.02662

Paul Pu Liang, Zhun Liu, Yao-Hung Hubert Tsai, Qibin Zhao, Ruslan Salakhutdinov, and Louis-Philippe Morency. 2019. Learning Representations from Imperfect Time Series Data via Tensor Rank Regularization. arXiv:1907.01011 [cs.LG]

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. 2015. Microsoft COCO: Common Objects in Context. arXiv:1405.0312 [cs.CV]

Zhun Liu, Ying Shen, Varun Bharadhwaj Lakshminarasimhan, Paul Pu Liang, Amir Zadeh, and Louis-Philippe Morency. 2018. Efficient Low-rank Multimodal Fusion with Modality-Specific Factors. arXiv:1806.00064 [cs.AI]

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks. arXiv:1908.02265 [cs.CV]

Jiasen Lu, Vedanuj Goswami, Marcus Rohrbach, Devi Parikh, and Stefan Lee. 2020. 12-in-1: Multi-Task Vision and Language Representation Learning. arXiv:1912.02315 [cs.CV]

Lin Ma, Zhengdong Lu, Lifeng Shang, and Hang Li. 2015. Multimodal Convolutional Neural Networks for Matching Image and Sentence. arXiv:1504.06063 [cs.CV]

Iain Matthews, Tim Cootes, J. Andrew Bangham, Stephen Cox, and Richard Harvey. 2002. Extraction of Visual Features for Lipreading. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24 (2002), 2002.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *Neural and Information Processing System (NIPS)*. https://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf

J. Ngiam, A. Khosla, Mingyu Kim, Juhan Nam, H. Lee, and A. Ng. 2011. Multimodal Deep Learning. In *ICML*.

Z. Niu, M. Zhou, L. Wang, X. Gao, and G. Hua. 2017. Hierarchical Multimodal LSTM for Dense Visual-Semantic Embedding. In *2017 IEEE International Conference on Computer Vision (ICCV)*. 1899–1907. https://doi.org/10.1109/ICCV.2017.208

Sunghyun Park, Han Suk Shim, Moitreya Chatterjee, Kenji Sagae, and Louis-Philippe Morency. 2014. Computational Analysis of Persuasiveness in Social Multimedia: A Novel Dataset and Multimodal Prediction Approach. In *Proceedings of the 16th International Conference on Multimodal Interaction* (Istanbul, Turkey) *(ICMI '14)*. Association for Computing Machinery, New York, NY, USA, 50–57. https://doi.org/10.1145/2663204.2663260

E. K. Patterson, S. Gurbuz, Z. Tufekci, and J. N. Gowdy. 2002. CUAVE: A new audio-visual database for multimodal human-computer interface research. In *In Proc. ICASSP*. 2017–2020.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global Vectors for Word Representation.. In *EMNLP*, Vol. 14. 1532–1543.

Juan-Manuel Pérez-Rúa, Valentin Vielzeuf, Stéphane Pateux, Moez Baccouche, and Frédéric Jurie. 2019. MFAS: Multimodal Fusion Architecture Search. arXiv:1903.06496 [cs.LG]

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. 2015. ImageNet Large Scale Visual Recognition Challenge. arXiv:1409.0575 [cs.CV]

Gaurav Sahu and Olga Vechtomova. 2021. Adaptive Fusion Techniques for Multimodal Data. arXiv:1911.03821 [cs.CL]

Heng Tao Shen, Luchen Liu, Yang Yang, Xing Xu, Zi Huang, Fumin Shen, and Richang Hong. 2020. Exploiting Subspace Relation in Semantic Labels for Cross-modal Hashing. *IEEE Transactions on Knowledge and Data Engineering* PP, 99 (2020), 1–1. https://doi.org/10.1109/tkde.2020.2970050

Haoyue Shi, Jiayuan Mao, Tete Xiao, Yuning Jiang, and Jian Sun. 2018. Learning Visually-Grounded Semantics from Contrastive Adversarial Samples. arXiv:1806.10348 [cs.CL]

Nitish Srivastava and Ruslan Salakhutdinov. 2012. Multimodal Learning with Deep Boltzmann Machines. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 2* (Lake Tahoe, Nevada) *(NIPS'12)*. Curran Associates Inc., Red Hook, NY, USA, 2222–2230.

Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. 2019. A Corpus for Reasoning About Natural Language Grounded in Photographs. arXiv:1811.00491 [cs.CL]

Sunny Verma, Jiwei Wang, Zhefeng Ge, Rujia Shen, Fan Jin, Yang Wang, Fang Chen, and Wei Liu. 2020. Deep-HOSeq: Deep Higher Order Sequence Fusion for Multimodal Sentiment Analysis. arXiv:2010.08218 [cs.AI]

Hao Wu, Jiayuan Mao, Yufeng Zhang, Yuning Jiang, Lei Li, Weiwei Sun, and Wei-Ying Ma. 2019. UniVSE: Robust Visual Semantic Embeddings via Structured Semantic Representations. arXiv:1904.05521 [cs.CV]

Fanyi Xiao, Leonid Sigal, and Yong Jae Lee. 2017. Weakly-supervised Visual Grounding of Phrases with Linguistic Structures. arXiv:1705.01371 [cs.CV]

Xitong Yang, Palghat Ramesh, Radha Chitta, Sriganesh Madhvanath, Edgar A. Bernal, and Jiebo Luo. 2017. Deep Multimodal Representation Learning from Temporal Data. arXiv:1704.03152 [cs.CV]

Amir Zadeh, Minghai Chen, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. 2017. Tensor Fusion Network for Multimodal Sentiment Analysis. arXiv:1707.07250 [cs.CL]

Amir Zadeh, Paul Pu Liang, Navonil Mazumder, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. 2018. Memory Fusion Network for Multi-view Sequential Learning. arXiv:1802.00927 [cs.LG]

Amir Zadeh, Rowan Zellers, Eli Pincus, and Louis-Philippe Morency. 2016. MOSI: Multimodal Corpus of Sentiment Intensity and Subjectivity Analysis in Online Opinion Videos. arXiv:1606.06259 [cs.CL]

Konstantinos Zagoris, Avi Arampatzis, and Savvas A. Chatzichristofis. 2010. Www.MMRetrieval.Net: A Multimodal Search Engine. In *Proceedings of the Third International Conference on SImilarity Search and APplications* (Istanbul, Turkey) *(SISAP '10)*. Association for Computing Machinery, New York, NY, USA, 117–118. https://doi.org/10.1145/1862344.1862363

Haien Zeng, Hanjiang Lai, Hanlu Chu, Yong Tang, and Jian Yin. 2019. Modal-aware Features for Multimodal Hashing. *arXiv* (2019). arXiv:1911.08479