International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Comparative Analysis of Various Textual-Visual Models for Self-Attentive Query Focused Video Summarization

Sheetal Girase[1*], Mangesh Bedekar[1], Devashish Bote[1], Vidya Dhopate[1]

Submitted: 03/02/2024 **Revised**: 11/03/2024 **Accepted**: 17/03/2024

Abstract: The exponential growth of video data presents a significant challenge in extracting pertinent information from it. Video summarization aims to address this issue by extracting essential information from video data in order to facilitate the exploration of videos. Given the subjective nature of determining "relevant information" in a video based on user preferences, it is imperative to establish a mechanism that takes into account the users' preferences during the process of generating a summary. One approach that can be employed is to enable users to input a query. Rather than generating a predetermined and inflexible summary for a given video input, this study has investigated a method of generating a video summary that caters to the preferences of the user. Query Focused Video Summarization (QFVS) is regarded as a supervised learning problem in the context of the YouTube Dataset [4]. It aims to produce a summary based on user inputs, specifically the video and the textual query. The query relevance of frames from the video is determined by mapping them to a shared multimodal semantic embedding space. By utilising our attention network and encoder, we have successfully enhanced the accuracy rate from 61.91% [4] to 74.60%. Extensive experiments were conducted utilising deep learning models, specifically ResNet34 and DenseNet, to extract image features. Additionally, word2vec and GloVe were employed for word mappings. The integration of textual and image features is employed for diverse experimental purposes.

Keywords: *Video summarization, keyframes, multimodal fusion, semantic embedding space*

1. Introduction

The proliferation of video data has presented a significant obstacle in the extraction of information due to the continuous expansion of video availability. The process of navigating this information presents a complex and intricate challenge. Furthermore, the definition of "important" in a generated summary is subjective and can vary among users. Therefore, an inflexible and unalterable summary is an inadequate resolution. In the context of conventional video summarization techniques, it is imperative to develop a mechanism that enables users to exert influence over the generated summary. The mechanism in question refers to a textual query that can be input by the user into the system. The term "QFVS" is used in this context [6], and we present a system that offers the user a customised summary based on their query, as depicted in Figure 1. In order to enhance user engagement in the process of summarization decision-making, several studies [1, 2, 4, 6, 7, 11, 12, 13, 16] have implemented a query-based approach. This approach enables users to input textual queries to retrieve relevant information from the video. Researchers have approached the task of video summarization by employing different models, such as supervised, unsupervised, and weakly supervised methods, based on the specific requirements of their problem formulation. The QFVS (Query-Focused Video Summarization) task is approached as a supervised learning problem in this study. The YouTube Links dataset, which was originally presented in [4], is utilised for this purpose. The dataset has been made publicly accessible to encourage further research in the field.

The superiority of QFVS over conventional video summarization techniques has been empirically established [7]. Despite the numerous benefits and enhanced functionality, the incorporation of a textual input feature for users also presents challenges in the realm of multimodal fusion. Specifically, these challenges pertain to training the summarizer model in a manner that effectively maps the textual query to the input video, thereby generating a comprehensive summary of the video content. Traditional video summarization methods typically generate a single fixed summary, as these models are trained solely on the video input without considering any other types of information. Therefore, it is imperative to comprehend and construct a conceptual framework that elucidates the correlation between the input query and the video in the context of Query-Focused Video Summarization (QFVS). By utilising this correlation, it becomes possible to compute the score of the frames by considering their query relevance. Consequently, the significant frames can be identified and selected in order to generate a summary that is pertinent to the given query.

[1]*Dr. Vishwanath Karad MIT World Peace University, Pune 411038, Maharashtra, India*
**Corresponding Author: Sheetal Girase*
**Dr. Vishwanath Karad MIT World Peace University, Pune 411038, Maharashtra, India*

Quantum Field Theory (QFT) has numerous applications across various domains, including but not limited to healthcare and sports. The capacity to comprehend the contextual nuances of a query and subsequently customise a video summary can prove advantageous in the contemporary digital landscape. Quantum factorization algorithms (QFAs) have the potential to be effectively trained using sports datasets, enabling their application in the analysis and strategic planning of games. For instance, by employing these models on datasets pertaining to cricket, they can be utilised to examine the instances when players successfully take wickets. Monitoring animal behaviours and tracking their movements in wildlife sanctuaries pose challenges due to the extensive duration of video surveillance footage. Quantitative field video surveillance (QFVS) can be employed in order to investigate various animal behaviours in that particular context. Additionally, this technology can be applied in the field of traffic surveillance to effectively monitor and detect any abnormal or atypical activities within extensive surveillance footage. Therefore, the refinement of QFVS can be achieved by utilising datasets that are specifically tailored to the application at hand, thereby enhancing its effectiveness.

This paper presents the empirical findings of our comprehensive solution proposed for video summarization, which is predicated on text queries provided. The justification for our proposed architecture is presented in section 3. In this study, we investigate various approaches to enhance the precision of our measurements by employing a rigorous experimental design. The detailed description of this setup can be found in sections 4.2 and 4.3. Section 5 encompasses the execution of output visualisation and metric evaluation for the model. In conclusion, the aforementioned contributions have been made.

- This study proposes an end-to-end model that utilises self-attention and deep learning techniques to learn both textual and visual embeddings for query-based video summarization.
- This study presents a comprehensive experimental analysis that investigates the impact of employing an attention network and a query modelling encoder on the quality of generated video summaries. The evaluation is conducted using the dataset introduced in reference [4].
- This study presents a comprehensive analysis of experimental outcomes aimed at evaluating the efficacy of visual feature extraction models, specifically DenseNet and ResNet34, in comparison to textual feature extraction models, namely word2vec and GloVe, for the task of Question-Focused Visual Search (QFVS).

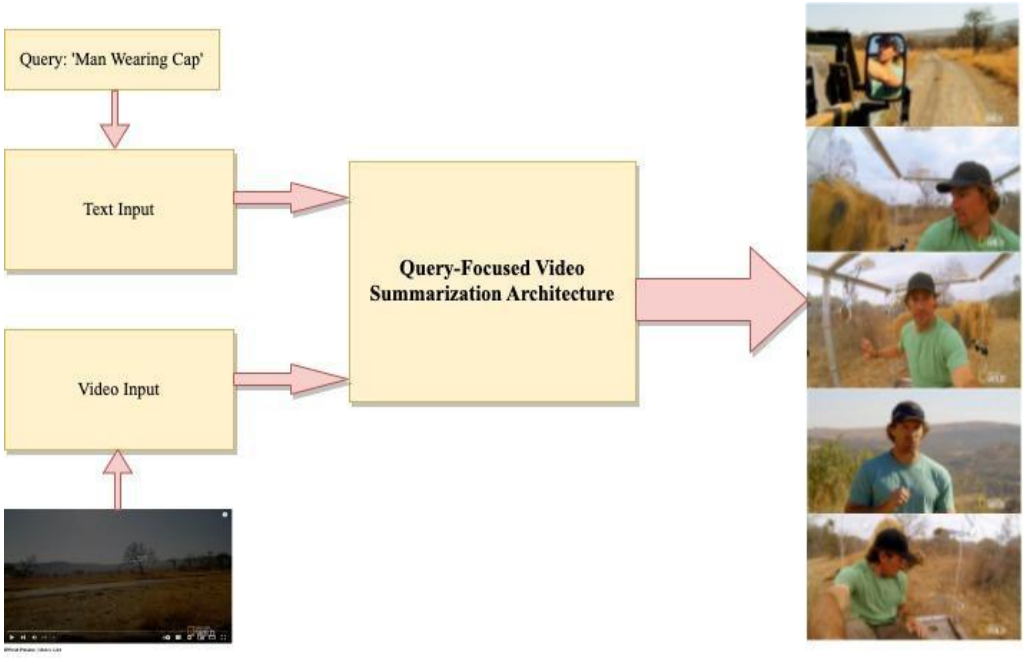

Figure 1 Overview of the QFVS process.

2. Literature Survey

The overarching subject of video summarization can be categorised in various ways based on the specific problem being examined. For instance, the utilisation of abstractive and extractive approaches has been discussed in previous literature [7]. The process of abstractive summarization involves converting the initial video content into a condensed and visually pleasing representation. For the purpose of this study, video trailers [14] and video synopses [12] were examined. In contrast, extractive methods are utilised to choose a subset of keyframes that contain pertinent information

[7, 8]. A commonly used categorization of video summarization techniques includes supervised learning approaches, as exemplified by previous works [1, 2, 4, 7, 9, 11, 13, 24-28], and unsupervised learning approaches, as demonstrated by prior studies [31-38]. There is a growing interest in the utilisation of semi-supervised or weakly supervised methodologies, as evidenced by the work discussed in reference [16]. The extraction of keyframes from an input video can be modelled using adversarial methods [3, 9, 11, 29, 30]. This modelling approach involves the interplay between a generator and a discriminator, which aims to enhance the robustness of the training model. Unsupervised methods for video summarization employ distinct attributes or properties as selection criteria and rely on manually designed heuristics to evaluate various aspects such as interestingness, diversity, and representativeness through the utilisation of low-level video features. Due to this factor, unsupervised learning exhibits strong performance when applied to problem statements that are specific to certain applications, but encounters difficulties in achieving generalisation.

Supervised learning models are trained using meticulously labelled training datasets that comprise videos to be summarised, along with corresponding ground-truth video summaries. The QFVS can be regarded as an extended problem of supervised video summarization. The utilisation of a supervised learning methodology, although constrained by the accessibility of accurately annotated datasets, facilitates the construction of robust end-to-end models. Therefore, in the context of our problem, we determine that supervised learning is the most suitable approach to employ. The methodology employed in our study bears the closest resemblance to the methodologies described in references [4] and [7]. The architectural framework presented in reference [4] shares similarities with our proposed architecture in terms of the overall flow of the end-to-end model. However, there exist notable distinctions in the specific methodologies employed to attain this flow. Additionally, our model is based on existing literature that supports the effectiveness of incorporating attention networks in video summarization models. For example, the authors in references [24, 29] have achieved competitive outcomes by incorporating attention mechanisms into their model. Additionally, reference [31] suggests a novel approach based on pure attention networks to address the challenges associated with the architectural complexity of B-RNN architectures.

3. Methodology

In our methodology, a keyframe refers to a frame that is deemed more appropriate in relation to the provided textual query. The input consists of a specific text query and a video. A mapping process is applied to associate the text query with the video, resulting in the selection of a group of keyframes. These keyframes are chosen based on their relevance to the query, and they are used to create a summary of the video. The proposed methodology comprises four primary components, namely the query input engine, video processing module, frame score generator, and summary decision module, as illustrated in Figure 2. The intricate functioning of these components is illustrated in Figure 3. The query input engine and video processing module are responsible for receiving user inputs, specifically the user query and video, and performing preprocessing tasks on them. The inputs that have undergone preprocessing are provided to the module responsible for generating frame scores. This module calculates the query relevance score for each frame. The summary decision module ultimately chooses the keyframes for the summary by considering their query relevance score.

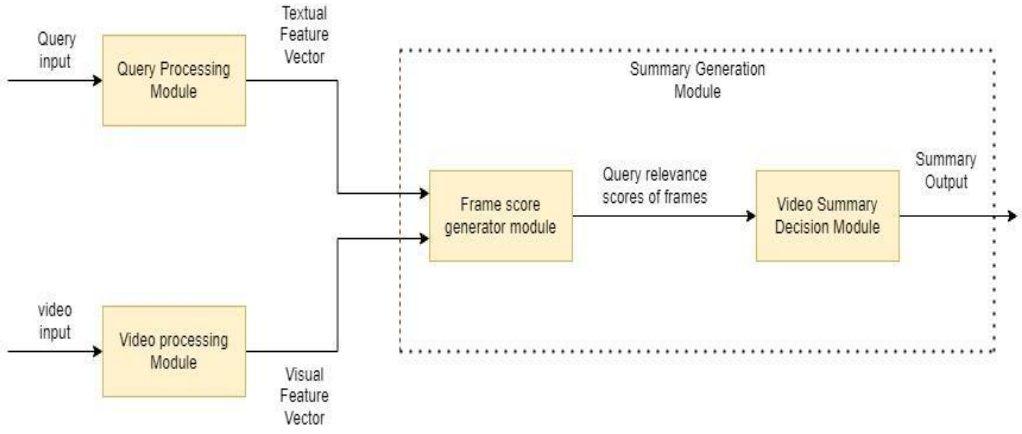

Figure 2 Components of Query-Focused Video Summarizer

Query Processing Module:

The proposed methodology aims to generate a comprehensive video summary solely based on the relevance of the query. The purpose of this module is to receive a user query Q as input and transform it into an encoded vector format, known as a textual feature vector

q'. The query Q is processed using the word2vec model, which has been trained on the Google News Dataset [19], in order to assign a 300-dimensional semantic representation to each word in the query, drawing inspiration from the research conducted by [8]. Moreover, the individual encoded vectors are subsequently fed into an encoder to collectively process the query and generate a fixed-sized representation of 512 dimensions for each query, referred to as the textual feature vector q'. The semantic meaning conveyed by the given query input is represented by the textual feature vector q'. The performance of the GloVe model as an alternative to the word2vec model has also been documented in section 5.

Video Processing Module:

The purpose of this component is to receive a video as input and convert it into frames V, which can then be further processed into the visual feature vector v. The initial step involves pre-processing each video at a rate of 1 frame per second (1fps) in order to convert it into individual frames, denoted as V. The frames are represented using a CNN-based pre-trained ResNet34 network [21]. The first 33 layers of the ResNet34 architecture are utilised to extract the feature vectors v' from the frames. A feature vector with a dimensionality of 512 is obtained by extracting features from the layer located immediately prior to the classification layer in the ResNet34 model. The symbol "v'" is employed to denote the information contained within the frames. Additionally, it serves the purpose of assessing its pertinence to the corresponding query. DenseNet [40] is employed as a convolutional neural network (CNN) based substitute for ResNet34, and a comparative analysis is provided in section 5.

Frame Score Generator Module:

The frame score generator, depicted in Figure 2, receives textual and visual feature vectors, denoted as q' and v', respectively, from the preceding two modules. These vectors are then processed by the attention mechanism. The final feature vectors q and v were derived from the attention scores obtained for each of the feature vectors. The attention network facilitates the frame score generator module in discerning the salient components within the provided inputs. The activation function is utilised to compute the final feature vectors q and v based on the input feature vectors q' and v'.

$$f(U,K) = SoftMax(U,K) \quad (1)$$

The weight matrices in the network are denoted by U and K. The function g is employed to determine the attention score prior to applying the SoftMax function to it. Typically, the implementation of g can involve the utilisation of diverse operations, including Multi-Layer Perceptron, dot product, and scaled dot product. Upon careful examination, the dot product was employed as the methodology for our approach. The calculation of the final feature vectors is performed after the acquisition of the attention scores.

$$c' = \sum_i^N f_i(U,K)B_i \quad (2)$$

In this context, the length of the input vector is represented by the variable N. The attention scores obtained are denoted as c', and Bi refers to the ith value in the weight matrix of the network. Additionally, the vector c' undergoes processing in the fully connected layer, which includes the incorporation of bias, and is subsequently subjected to dropout.

$$\boldsymbol{c} = dropout(Wc' + x) \quad (3)$$

The given equation represents a neural network, where the weight matrix is symbolised by W, the bias term is represented by x, and c represents the final feature output along with the attention values. Let c be an element belonging to the set {q, v}. Hence, the computation of the ultimate textual and visual feature vectors, denoted as q and v, is accomplished through the utilisation of the attention network.

Once the final feature vectors q and v are obtained, the frame score generator module proceeds to rank the frames according to their relevance to the query. The frame score generator aims to establish a connection between the textual and visual feature vectors by mapping them in a shared textual-visual semantic embedding space [41]. After undergoing training, the calculation of the equivalence between the features can be accomplished by utilising the cosine similarity equation provided below.

$$s(q,v) = \frac{q.v}{|q||v|} \quad (4)$$

The utilisation of cosine similarity facilitates the evaluation of the degree of semantic proximity between each frame and the provided query. The network is trained with the primary goal of meeting the rank constraint. This constraint ensures that, when presented with a query Q, the relevance score of the relevant frames V+ is greater than the relevance score of the irrelevant frames V-. This objective is outlined in references [7, 22].

$$s(q,v^+) = s(q,v^-) \quad (5)$$

The similarity score is subsequently propagated through the network, wherein every input node is connected to every output node, resulting in a final query relevance score for the frame that spans from 0 to 3. In order to facilitate the network's acquisition of this constraint and facilitate the training of the model, we employed the Cross-Entropy loss function in the following manner:

$$Loss(y,\ y_{actual}) = -y[y_{actual}\,] + \ln(\sum_{j=1}^{M} \exp(y[j])(6)$$

Let y_actual represent the true class label, y represent the predicted class label, and M represent the total number of samples. The Adam optimizer [23] is utilised in this study, with the optimizer parameters set as β1 = 0.9 and β2 = 0.999. In order to enhance the numerical stability, a parameter denoted as ε is employed, with a value of 1e-8. In order to train our network, a learning rate of 1e-4 is utilised in conjunction with L2 normalisation to mitigate the risk of overfitting. Additionally, a batch size of 10 is employed.

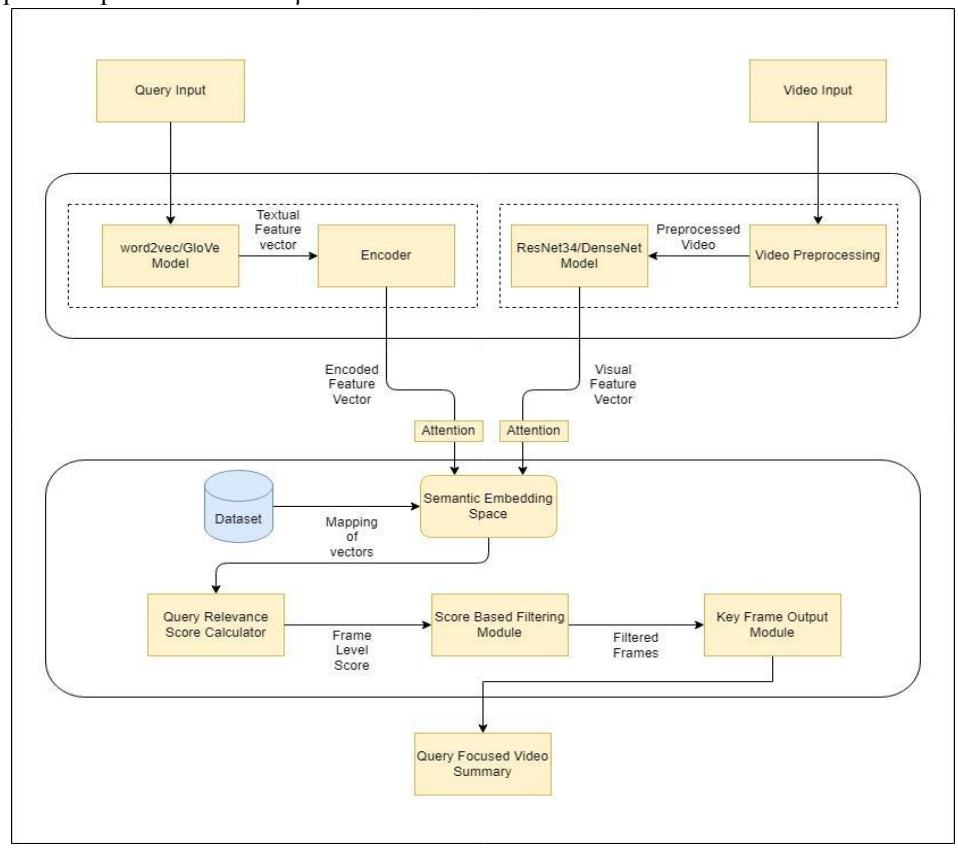

Figure 3 Detailed overview of Query-Focused Video Summarizer

Video Summary Decision Module:

The frame score generator assigns scores to frames ranging from 0 to 3, reflecting their relevance to the given query. A score of 0 indicates very poor relevance, while a score of 3 indicates very high relevance, as defined in reference [4]. The purpose of this module is to apply a filtering process to the frames, taking into consideration their query relevance score, in order to produce a summary. The process involves filtering out all frames that have been assigned a score of 2 or higher, followed by sorting these frames based on the probability assigned to them by the model. The algorithm subsequently identifies the most salient K frames, which are then utilised to construct the ultimate summary. The value of K is determined by the user, as it defines the desired size of the summary.

4. Experiment

This section provides an explanation of the experimental setup and dataset utilised in this approach. In the preceding section, a comprehensive elaboration of the evaluation metrics has been presented. In order to comprehend the influence of different query-processing modules, we have alternatively employed word2vec and GloVe. In a similar vein, we have conducted experiments utilising ResNet34 and DenseNet architectures to handle the visual features within the model. The objective of this study is to analyse and evaluate the effects of various textual and visual feature extraction modules on query-focused video summarization.

4.1 Dataset

Experiments were conducted on the YouTube video-based dataset for QFVS, as introduced in reference [4], while adhering to the RAD dataset outlined in reference [7]. The dataset comprises 190 YouTube videos that were obtained through text queries. The content of these YouTube videos is derived from the analysis of popular search queries on YouTube spanning the period from 2008 to 2016. Furthermore, these videos have been categorised into 22 distinct and varied categories. The conversion process involves transforming each video into individual frames, with a frame rate of 1 frame per second (fps), resulting in a total of up to 199 video frames. In order to mitigate subjectivity, a query

relevance score is assigned to each frame by five distinct workers from Amazon Mechanical Turk (AMT). The workers of the Amazon Mechanical Turk (AMT) platform assign scores at the frame level, ranging from 0 to 3. These scores are mapped to corresponding qualitative descriptors, where 0 represents "very bad," 1 represents "bad," 2 represents "good," and 3 represents "very good." Subsequently, these ratings undergo manual verification in order to mitigate potential errors. Every element within the dataset comprises the URL of the image corresponding to the sampled frame from a YouTube video, the associated query, and its corresponding relevance score.

4.2 Experimental Setup

The division of training and testing data has been allocated in an 80:20 ratio. In our study, a training dataset consisting of 152 videos was utilised to develop our model, while a separate testing dataset comprising a total of 38 videos was employed to evaluate its performance. The aforementioned data has been pre-processed in accordance with the previously outlined procedure for frame extraction. Each video is associated with a specific query, and the relevance of each frame is scored based on the query. The queries are restricted to a maximum of eight words. The model utilises the word2vec/GloVe model to encode each word in the query. The word2vec model produces a vector of 300 dimensions, whereas GloVe generates a vector of 200 dimensions. Next, the encoded words undergo the encoding process. In the case where there is no Encoder, we calculate the average of the encoded vectors. According to the specifications outlined in reference [4], the visual feature vector employs a convolutional neural network (CNN) input frame size of 224 by 224, consisting of three channels. Additionally, the process of normalising each image channel is performed. The model is trained by adjusting different hyperparameters, including the number of epochs, learning rate, and l2 normalisation. Empirical evidence for the optimal model is gathered by employing different combinations of these parameters. Moreover, the testing dataset is utilised to assess the trained models through 20 iterations, and the evaluation metrics are determined based on the highest value obtained from these iterations.

4.3 Evaluation Metrics

For comparison purpose, the evaluation metrics, Accuracy, Precision, Recall, and F1 score have been used. They are computed based on the predicted and actual query relevance scores of video frames as described below. Accuracy is defined as the percentage of correct predictions for the input from test data. F1 score computes how many times a model made a correct prediction across the entire dataset by combining precision and recall. Due to our biased nature of the dataset we are using it as the main evaluating factor for our model's performance.

$$Accuracy = \frac{Number\ of\ correct\ relevance\ score\ predictions}{Total\ number\ of\ samples} \quad (7)$$

$$Precision = \frac{Number\ of\ correctly\ predicted\ keyframes}{Total\ number\ of\ samples} \quad (8)$$

$$Recall = \frac{Number\ of\ correctly\ predicted\ keyframes}{Number\ of\ actual\ keyframes\ in\ dataset} \quad (9)$$

$$F1\ Score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (10)$$

5. Result and Analysis

This discussion pertains to the examination and evaluation of both quantitative and qualitative analysis methodologies. We assess and analyse our experimental findings and outcomes by utilising the evaluation metrics previously mentioned. The objective of this study is to identify the areas that require improvement in processing textual, visual, and multimodal inputs. To achieve this, we utilised both the word2vec and GloVe models in conjunction with the ResNet34 and DenseNet architectures. In the context of multimodal fusion, we have successfully implemented both the encoder and attention network. Subsequently, we proceeded to compare their respective quantitative outcomes, which are presented below.

Quantitative Results

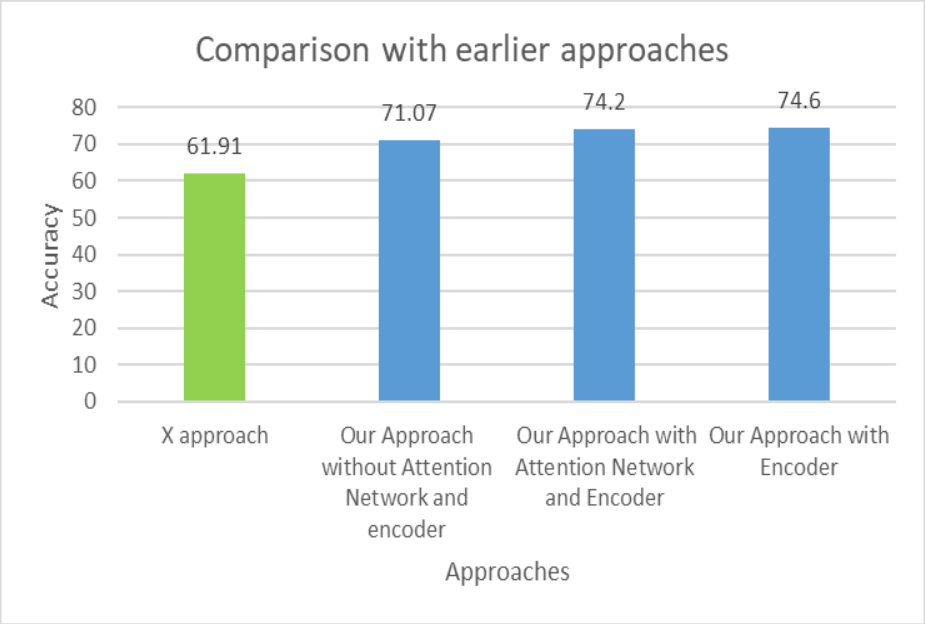

Figure 4 Comparison with the earlier approach

Our multimodal video summarization model with attention network and encoder was developed based on the research conducted by Jia-Hong Huang [4]. The accuracy of the different model architectures was compared with Jia-Hong Huang [4], as shown in Figure 4. The inclusion of attention networks, as well as the utilisation of encoders, in our methodologies have resulted in notable enhancements in accuracy. Figure 5 displays the training accuracy and loss graphs of our highest-performing model. Based on the characteristics exhibited by the graphs, it can be concluded that the training samples offered by the YouTube dataset [4] are adequate for the acquisition of knowledge by our model.

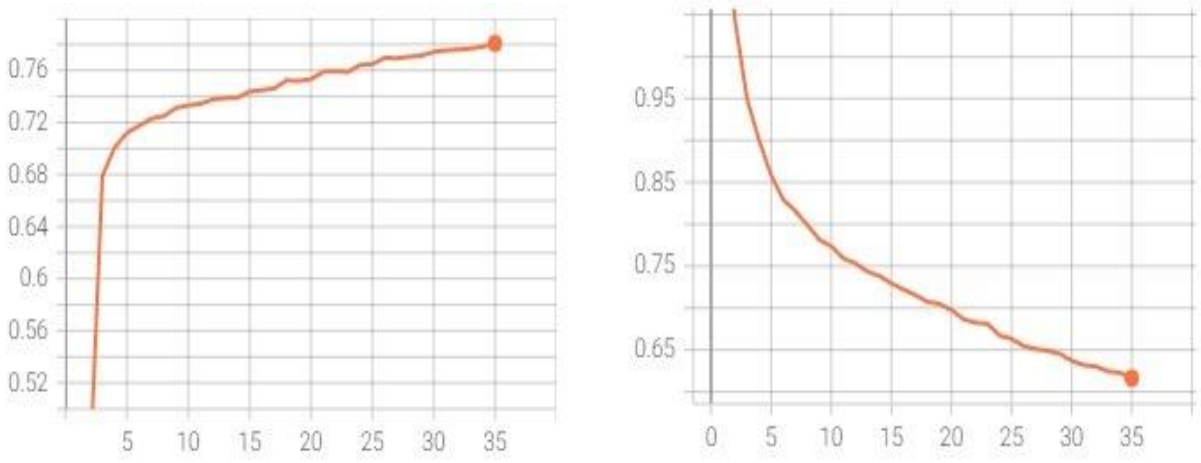

Fig. 5 Training accuracy and loss graphs

A series of experiments were conducted to explore various combinations for our approach, as presented in Table 1. The utilisation of an attention network in conjunction with an encoder yields superior outcomes in terms of the extracted image features from ResNet34 and text features from word2vec. The observed enhancement in the outcomes of this architectural model may be attributed to the incorporation of an attention network and encoder for multi-modal semantic embedding, in conjunction with the utilisation of ResNet34 and word2vec for feature extraction.

Table 1 Comparison of different model configurations.

Image Processing Model	Text Processing Model	Attention Network	Encoder	Accuracy	Precision	Recall	F1-Score
Resnet34	word2vec	✓	✓	**74.20**	**64.81**	**74.20**	**69.10**
			✓	**74.60**	**63.18**	**74.60**	**67.72**
				71.07	57.43	71.07	63.44
	GloVe	✓	✓	**74.20**	**64.85**	**74.20**	**69.07**
			✓	75.78	68.94	75.78	65.46
				75.71	59.29	75.71	65.30
Densenet	word2vec	✓	✓	71.00	61.27	71.00	65.67
			✓	73.78	62.16	73.78	66.89
				70.91	60.35	70.91	64.45
	GloVe	✓	✓	72.99	60.88	72.99	65.90
			✓	73.97	61.19	73.97	66.19
				74.85	63.02	74.85	67.53

Table 1 showcases the three most proficient model architectures based on our findings. The best-performing models have been marked bold in the table. The findings of our study indicate that the utilisation of DenseNet for image feature representation in model architectures is observed to exhibit lower performance in comparison to model architectures that employ ResNet34. Our hypothesis posits that the excessive complexity of the DenseNet model architecture may be the cause of reduced interpretability. The findings additionally indicate that word2vec outperforms GloVe in relation to this particular task. The three model architectures we have chosen as our top performers incorporate an encoder specifically designed for the purpose of creating a multimodal semantic embedding space.

Qualitative Results

In this subsection, we demonstrate Qualitative Analysis for our best-performing models in **Figures 6, 7, and 8**. For validation purposes, we have used unseen video from YouTube as input for all the approaches with the query input as *'Lion Running'*. This video taken from YouTube is a trailer of a National Geographic show. For the comparison and representation, we have shown K=5 i.e. 5 keyframes to represent the summary below.

Query: 'Lion Running'

Figure 6 Qualitative Summary generated by Model Architecture ResNet34 + word2vec + attention + encoder

Figure 7 Qualitative Summary generated by Model Architecture ResNet34 + GloVe + attention + encoder

Figure 8 Qualitative Summaries generated by Model architecture: Resnet34 + GloVe + encoder

6. Conclusion

Here QFVS is treated as a supervised learning problem. To tackle this, we propose a multimodal semantic embedding technique for generating video summaries in the form of keyframes. We train different model architectures that consist of different combinations of feature extraction models, encoder, and attention network. The results of our experimental setup for these model architectures are compared in **Table 1.** Our results show that, using an encoder and an attention network for multimodal semantic embedding leads to a significant performance gain. Architectures employing ResNet34 outperform those employing DenseNet in our experiments. Our experiments also indicate that word2vec performs better than GloVe in our model architectures.

In the future, exploring our model architectures in the context of domain-specific datasets is an interesting prospect. Domain-specific datasets do have their complexities. To handle these complexities, it will be necessary to employ more sophisticated attention networks as compared to the one used here. This model can further be extended to handle more modalities like audio.

References

[1] Sharghi, Aidean, Boqing Gong, and Mubarak Shah. "Query-focused extractive video summarization." European Conference on Computer Vision. Springer, Cham, 2016.

[2] Sharghi, Aidean, Jacob S. Laurel, and Boqing Gong. "Query-focused video summarization: Dataset, evaluation, and a memory network based approach." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

[3] Plummer, Bryan A., Matthew Brown, and Svetlana Lazebnik. "Enhancing video summarization via vision-language embedding." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

[4] Huang, Jia-Hong, and Marcel Worring. "Query-controllable video summarization." Proceedings of the 2020 International Conference on Multimedia Retrieval. 2020.

[5] Ajmal, Muhammad, et al. "Video summarization: techniques and classification." International Conference on Computer Vision and Graphics. Springer, Berlin, Heidelberg, 2012.

[6] Xiao, Shuwen, et al. "Query-biased self-attentive network for query-focused video summarization." IEEE Transactions on Image Processing 29 (2020): 5889-5899.

[7] Vasudevan, Arun Balajee, et al. "Query-adaptive video summarization via quality-aware relevance estimation." Proceedings of the 25th ACM international conference on Multimedia. 2017.

[8] Lee, Y. J., Ghosh, J., & Grauman, K. (2012, June). "Discovering important people and objects for egocentric video summarization." In 2012 IEEE conference on computer vision and pattern recognition (pp. 1346-1353). IEEE.

[9] Gygli, Michael, Helmut Grabner, and Luc Van Gool. "Video summarization by learning submodular mixtures of objectives." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.

[10] Li, Sheng, et al. "Visual to text: Survey of image and video captioning." IEEE Transactions on Emerging Topics in Computational Intelligence 3.4 (2019): 297-312.

[11] Zhang, Yujia, et al. "Query-conditioned three-player adversarial network for video summarization." arXiv preprint arXiv:1807.06677 (2018).

[12] Ahmed, Sekh Arif, et al. "Query-based video synopsis for intelligent traffic monitoring applications." IEEE Transactions on Intelligent Transportation Systems 21.8 (2019): 3457-3468.

[13] Ji, Zhong, et al. "Query-aware sparse coding for multi-video summarization." arXiv preprint arXiv:1707.04021 (2017).

[14] Oosterhuis, Harrie, Sujith Ravi, and Michael Bendersky. "Semantic video trailers." arXiv preprint arXiv:1609.01819 (2016).

[15] Sreenu, G., and MA Saleem Durai. "Intelligent video surveillance: a review through deep learning techniques for crowd analysis." Journal of Big Data 6.1 (2019): 1-27.

[16] Mithun, Niluthpol Chowdhury, Sujoy Paul, and Amit K. Roy-Chowdhury. "Weakly supervised video moment retrieval from text queries." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019.

[17] Del Molino, Ana Garcia, et al. "Summarization of egocentric videos: A comprehensive survey." IEEE

Transactions on Human-Machine Systems 47.1 (2016): 65-76.
[18] Baskurt, Kemal Batuhan, and Refik Samet. "Video synopsis: A survey." Computer Vision and Image Understanding 181 (2019): 26-38.
[19] Mikolov, Tomas, et al. "Distributed representations of words and phrases and their compositionality." Advances in neural information processing systems. 2013.
[20] Sebastian, Tinumol, and Jiby J. Puthiyidam. "A survey on video summarization techniques." Int. J. Comput. Appl 132.13 (2015): 30-32.
[21] He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
[22] Frome, Andrea, Greg Corrado, Jonathon Shlens, Samy Bengio, Jeffrey Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. "DeViSE: A deep visual-semantic embedding model." (2013).
[23] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." arXiv preprint arXiv:1412.6980 (2014).
[24] Xiao, Shuwen, et al. "Convolutional hierarchical attention network for query-focused video summarization." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 34. No. 07. 2020.
[25] Sharghi, Aidean, et al. "Improving sequential determinantal point processes for supervised video summarization." Proceedings of the European Conference on Computer Vision (ECCV). 2018.
[26] Zhang, Ke, et al. "Video summarization with long short-term memory." European conference on computer vision. Springer, Cham, 2016.
[27] Gong, Boqing, et al. "Diverse sequential subset selection for supervised video summarization." Advances in neural information processing systems 27 (2014): 2069-2077.
[28] Zhang, Ke, et al. "Summary transfer: Exemplar-based subset selection for video summarization." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
[29] Fu, Tsu-Jui, Shao-Heng Tai, and Hwann-Tzong Chen. "Attentive and adversarial learning for video summarization." 2019 IEEE Winter Conference on Applications of Computer Vision (WACV). IEEE, 2019.
[30] Zhang, Yujia, et al. "DTR-GAN: Dilated temporal relational adversarial network for video summarization." Proceedings of the ACM Turing Celebration Conference-China. 2019.
[31] Fajtl, Jiri, Hajar Sadeghi Sokeh, Vasileios Argyriou, Dorothy Monekosso, and Paolo Remagnino. "Summarizing videos with attention." In Asian Conference on Computer Vision, pp. 39-54. Springer, Cham, 2018.
[32] Chu, Wen-Sheng, Yale Song, and Alejandro Jaimes. "Video co-summarization: Video summarization by visual co-occurrence." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.
[33] De Avila, Sandra Eliza Fontes, Ana Paula Brandao Lopes, Antonio da Luz Jr, and Arnaldo de Albuquerque Araújo. "VSUMM: A mechanism designed to produce static video summaries and a novel evaluation method." Pattern Recognition Letters 32, no. 1 (2011): 56-68.
[34] Ngo, Chong-Wah, Yu-Fei Ma, and Hong-Jiang Zhang. "Automatic video summarization by graph modeling." Proceedings Ninth IEEE International Conference on Computer Vision. IEEE, 2003.
[35] Panda, Rameswar, and Amit K. Roy-Chowdhury. "Collaborative summarization of topic-related videos." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017.
[36] Zhu, Xiatian, Chen Change Loy, and Shaogang Gong. "Video synopsis by heterogeneous multi-source correlation." Proceedings of the IEEE International Conference on Computer Vision. 2013.
[37] Zhou, Kaiyang, Yu Qiao, and Tao Xiang. "Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 32. No. 1. 2018.
[38] Rochan, Mrigank, and Yang Wang. "Video summarization by learning from unpaired data." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019.
[39] Jeffrey P, Richard S, Christopher DM. "GloVe: global vectors for word representation." In: Proceedings of the empirical methods in natural language processing (EMNLP 2014) 12. 2014
[40] G. Huang, Z. Liu, L. Van Der Maaten and K. Q. Weinberger, "Densely Connected Convolutional Networks." 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2261-2269, doi: 10.1109/CVPR.2017.243.
[41] Visual-Semantic Alignment Across Domains Using a Semi-Supervised Approach