Focal Correlation and Event-based Focal Visual-Content Text Attention for Past Event Search

Pranita P. Deshmukh * and Shanmugam Poonkuntran

Posted Date: 23 May 2025

doi: 10.20944/preprints202505.1872.v1

Keywords: convolutional layer non-linearity; event-based focal visual-content text attention; focal correlation; long short-term memory; past event search; visual-content text attention; visual question answering

 doi:10.20944/preprints202505.1872.v1

Article

Focal Correlation and Event-Based Focal Visual-Content Text Attention for Past Event Search

Pranita P. Deshmukh * and Shanmugam Poonkuntran

VIT Bhopal University, Bhopal-Indore Highway Kothrikalan, Sehore 466114, Madhya Pradesh, India

* Correspondence: pranita.33deshmukh@gmail.com; Tel.: +91-94-2104-4780

Abstract: Every minute, countless hours of video and an enormous number of images are uploaded by people worldwide to the internet and social media platforms. Over a span of just a few years, users on these platforms, often utilizing their smartphones, accumulate tens of thousands of photos and countless hours of video. These recordings capture cherished moments such as weddings, family gatherings, and birthday parties, creating a vivid visual archive of human experiences and offering a unique window into contemporary life. When analyzed correctly, these images and videos can help reconstruct important events, from personal milestones to significant historical occurrences like war crimes, human rights violations, and terrorist acts. Each photo and video holds a wealth of information, often arranged in sequences based on timestamps and accompanied by text annotations, tags, or other metadata. However, handling this multimodal data presents challenges, particularly when some photos or videos lack annotations. A sophisticated method has been developed to manage these irregularities, ensuring relevant videos are displayed as evidence alongside direct answers derived from the input data. This system not only provides precise answers but also includes pertinent evidential photos or text snippets to substantiate the reasoning process. Examining every photo, video, or text query and answer manually is time-consuming. Therefore, it is crucial to identify relevant photos or videos quickly to verify answers efficiently. In this study, we propose a Long Short-Term Memory (LSTM) based visual question answering model integrated with convolutional layer non-linearity to tackle this challenge. The proposed adaptive memory network with attention mechanism model is designed and evaluated on common objects in context dataset with weight initialization and regularization, and model optimization. The Event-based Focal Visual-Content Text Attention (EFVCTA) framework is proposed in this paper. The system is designed on the basis of the proposed EFVCTA for the automated retrieval of historical events through the utilization of visual question answering (VQA) technology. The system can be used to identify the past event's information from the photos of the various training programmes, workshops, conferences, annual social gatherings, etc. conducted in the academic institutions.

Keywords: convolutional layer non-linearity; event-based focal visual-content text attention; focal correlation; long short-term memory; past event search; visual-content text attention; visual question answering

1. Introduction

A common way to recall past events or memories is through questioning, such as asking, "When did we last visit the garden?" Addressing this challenge, we present an innovative system designed for automatic retrieval of past events using visual question answering in collections of photographs or video footage. Our innovative system processes a sequence of questions or images from a user with the goal of automatically responding to inquiries about past events by providing relevant photos, videos, or responses. In our system, users can input either a question, a series of photos, or both. The system then produces an answer along with pertinent photos or videos that validate and substantiate the response. The inclusion of accompanying visuals is pivotal, serving not only to

 doi:10.20944/preprints202505.1872.v1

swiftly verify the answer but also to offer comprehensive details that aid in refreshing the user's memory regarding the event in question.

In the span of a few years, a regular smartphone user could accumulate hundreds of photos documenting vacations or special occasions. This accumulation eventually translates into tens of thousands of images and endless hours of video, capturing treasured memories like weddings, family reunions, and birthday celebrations. Research has shown that people often use personal photos and videos to recall memories of these events. The challenge, however, lies in identifying specific event-related photos or videos from this vast collection. This study addresses this concern by introducing a system designed for automated retrieval of past events through a visual question answering dataset. This system leverages advanced techniques to sift through extensive photo and video libraries, pinpointing the relevant visuals that correspond to specific memories and questions.

The outlined system for automated retrieval of past events through visual question answering is depicted in Figure 1. The process commences with the integration of a visual question answering (VQA) dataset into the system. Subsequently, a question understanding network model is devised to interpret user-submitted queries. The system searches the photo/video dataset based on the textual information extracted from the questions or the meaning-based features identified within them. Each query is addressed based on the retrieved images or videos associated with the relevant event. To ensure accuracy, the system generates evidence alongside the constructed answers, allowing users to verify the correctness of the responses. The system's efficacy is assessed through diverse metrics and is realized in Python, harnessing VQA and alternative QA datasets.

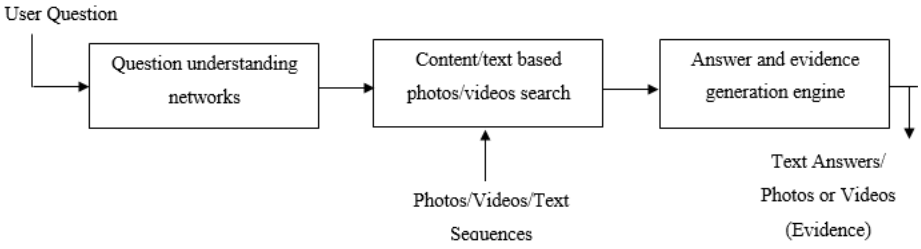

Figure 1. Proposed Methodology for Designing an Automatic Past Event Search System Using Visual Question Answering.

The research objectives are achieved through the proposal, design, and development of various models. In this work LSTM based VQA model which integrates convolutional layer non-linearity is proposed for addressing question understanding. An adaptive memory network is developed with attention mechanism which utilizes weight regularization and optimization for addressing content based and text based photos or videos search. A novel strategy event-based focal visual-content text attention (EFVCTA) model is proposed, designed and developed for question context identification, text embedding with sequence encoder, temporal focal correlation identification, and visual content-text attention for past event search with evidence generation. These contributions of this research work are discussed in the subsequent sections.

2. Long Short-Term Memory (LSTM) Based Visual Question Answering Model Integrating Convolutional Layer Non-Linearity

The Long Short-Term Memory (LSTM) network is a type of Recurrent Neural Network (RNN) architecture utilized in the realm of deep learning [1]. Distinguished from traditional feed-forward neural networks, LSTMs incorporate feedback connections, which empower them to handle not just isolated data points, like images, but also entire sequences of data, such as speech or video. This capability renders LSTMs apt for tasks including unsegmented, connected handwriting recognition [2], speech recognition [3,4], and the detection of anomalies in network traffic or intrusion detection systems (IDSs). An LSTM unit generally consists of a cell, an input gate, an output gate, and a forget gate. The cell maintains values over varying time periods, while the gates regulate the flow of

 doi:10.20944/preprints202505.1872.v1

information into and out of the cell. This architecture allows LSTMs to process sequential data and maintain their hidden state over time, making them ideal for analyzing, interpreting, and forecasting time series data, even when there are unpredictable delays between critical events. LSTMs were ingeniously crafted to overcome the vanishing gradient challenge that plagues the training of conventional RNNs. Their relative insensitivity to the length of gaps in the data gives LSTMs an edge over RNNs, hidden Markov models, along with various other sequence learning techniques, across numerous applications. Although classic RNNs theoretically possess the capability to capture long-term dependencies in input sequences, they encounter practical difficulties. During back-propagation, long-term gradients can either diminish to zero (vanish) or grow infinitely (explode) due to the limitations of finite-precision numbers in computations. LSTM units alleviate the vanishing gradient issue by permitting gradients to flow without alteration. Nonetheless, LSTM networks remain susceptible to the exploding gradient problem [5].

Standard Vanilla LSTMs struggle to model input with spatial structures, such as images. To overcome this limitation, the CNN Long Short-Term Memory Network (CNN LSTM) was introduced. Tailored for sequence prediction problems involving spatial inputs like images or videos, this architecture merges Convolutional Neural Network (CNN) layers for feature extraction with LSTMs for sequence prediction. CNN LSTMs were specifically developed to address visual time series prediction challenges and to generate textual descriptions from image sequences, such as videos. They excel in tasks including:

- Activity Recognition: Crafting a narrative to describe activities depicted in a series of images.
- Image Description: Formulating a textual summary for a single image.
- Video Description: Composing a narrative for a sequence of images.

By leveraging CNNs to extract features from the input data and LSTMs to handle sequence prediction, CNN LSTMs effectively bridge the gap between spatial input processing and sequence modeling.

CNN LSTMs are a versatile class of models, adept at handling both spatial and temporal complexities, making them ideal for a range of vision tasks involving sequential inputs and outputs. Initially known as Long-term Recurrent Convolutional Networks (LRCNs), we will refer to these models more generically as "CNN LSTMs" to denote LSTMs integrated with CNNs. This architecture excels at generating textual descriptions of images by leveraging a CNN pre-trained on a challenging image classification task, repurposed as a feature extractor for the caption generation problem. Using a CNN as an image "encoder" is a natural choice: the CNN is first pre-trained for image classification, and its last hidden layer is then used as input to the RNN decoder that generates sentences. This architecture transcends image captioning to encompass speech recognition and natural language processing, leveraging CNNs as powerful feature extractors for LSTMs that manage audio and textual input data.

CNN LSTMs are particularly suitable for problems that:

- Feature spatial structure in their input, like the 2D pixel layout of an image or the 1D sequence of words spanning a sentence, paragraph, or document.
- Include temporal structure in their input, such as image sequences in a video or text with sequential words, or necessitate generating outputs with temporal structure, such as words forming a textual description.

VGG16 is a landmark convolutional neural network model developed by K. Simonyan and A. Zisserman at the University of Oxford, detailed in their paper "Very Deep Convolutional Networks for Large-Scale Image Recognition". The model notably achieved a top-5 test accuracy of 92.7% on the ImageNet dataset, encompassing more than 14 million images spread across 1,000 classes. VGG16 was a notable submission to the ILSVRC-2014 competition. The key improvement VGG16 made over its predecessor, One key innovation in AlexNet was its adoption of multiple successive 3×3 kernel-sized filters, contrasting with the larger 11 and 5 kernel-sized filters employed in its initial convolutional layers. This architectural change enhanced the network's performance and allowed for

 doi:10.20944/preprints202505.1872.v1

deeper layers. VGG16 underwent extensive training spanning several weeks on NVIDIA Titan Black GPUs. The ImageNet dataset, a vast repository, encompasses over 15 million labeled high-resolution images across approximately 22,000 categories.

Images were collected from the web and annotated by human labelers using Amazon's Mechanical Turk crowd-sourcing tool. Since 2010, the Pascal Visual Object Challenge has hosted the annual ImageNet Large-Scale Visual Recognition Challenge (ILSVRC), focusing on a subset of ImageNet that includes approximately 1,000 images per category. Overall, ImageNet comprises about 1.2 million training images, 50,000 validation images, and 150,000 testing images. To standardize the dataset, which initially featured images of varying resolutions, all were resized to a fixed 256×256 resolution. This involved extracting a central 256×256 patch through resizing and cropping rectangular images. This standardization process ensures uniformity and consistency in the dataset, facilitating effective analysis and processing. Figure 2 illustrates the architectural layout of VGG16, a prominent convolutional neural network model.

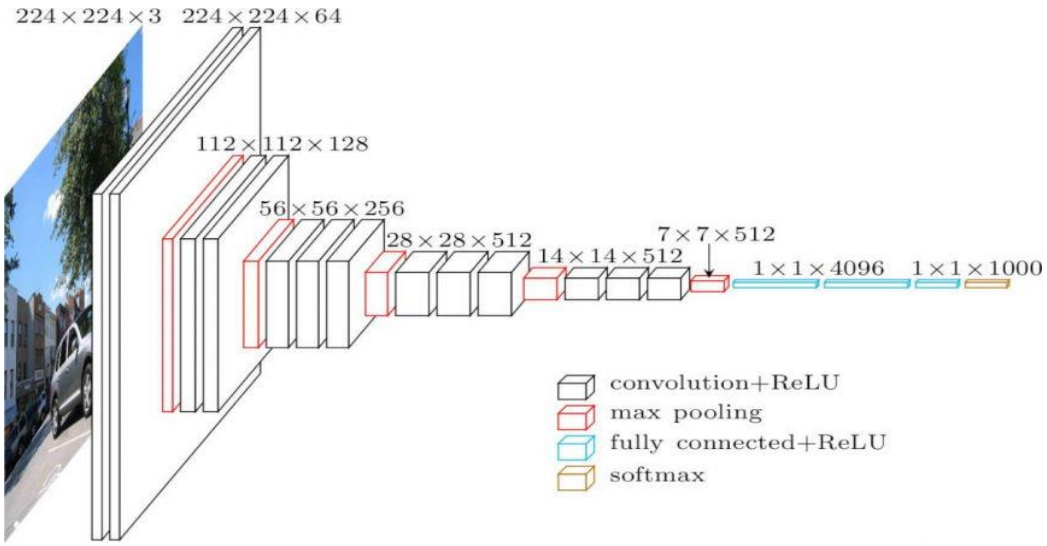

Figure 2. Architecture of VGG16.

The cov1 layer initially processes a standardized input size of 224 x 224 RGB image. This image undergoes transformation through a sequence of convolutional (conv.) layers, utilizing 3×3 filters that efficiently capture left/right, up/down, and central features. Additionally, certain configurations integrate 1×1 convolution filters, serving as linear transformations of input channels followed by non-linear operations. The convolutional stride remains consistent at 1 pixel, with spatial padding adjusted (typically 1-pixel padding for 3×3 conv. layers) to preserve spatial resolution post convolution. Spatial pooling occurs across five max-pooling layers, strategically interspersed within the convolutional layers—although not every conv. layer is followed by max-pooling. Max-pooling operates over 2×2-pixel windows with a stride of 2 pixels.

After the convolutional layers, the architecture progresses to three Fully-Connected (FC) layers, each with differing depths depending on the specific model design. The first two FC layers are equipped with 4096 channels each, ensuring robust feature representation. The third layer, pivotal for the 1000-way ILSVRC classification task, contains 1000 channels, corresponding to each class in the dataset. The final layer is dedicated to the soft-max operation. Consistency is maintained in the arrangement of fully connected layers across all network configurations.

All hidden layers incorporate rectification (ReLU) non-linearities. Interestingly, except for one network, none incorporate Local Response Normalization (LRN), as it does not improve performance on the ILSVRC dataset and adds to memory consumption and computational time. The ConvNet configurations, illustrated in Figure 3 and labeled from A to E, follow a uniform architectural blueprint, varying primarily in depth. Network A, for instance, includes 11 weight layers (comprising

 doi:10.20944/preprints202505.1872.v1

8 conv. and 3 FC layers), while Network E boasts 19 weight layers (with 16 conv. and 3 FC layers). The width of conv. layers (i.e., the number of channels) begins modestly at 64 in the initial layer and doubles progressively after each max-pooling stage, ultimately reaching 512 channels. This incremental widening allows for increasingly intricate feature extraction and representation as the network delves deeper into the hierarchy of features. The proposed LSTM based VQA model integrating convolutional layer non-linearity is shown in Figure 4. Anaconda with Python distribution is used to carry out the implementation. The libraries are used with python are: spaCy, OpenCV, Scikit-learn, Keras, TensorFlow, and VGG 16 Pre-trained Weights.

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

Figure 3. ConvNet Configurations.

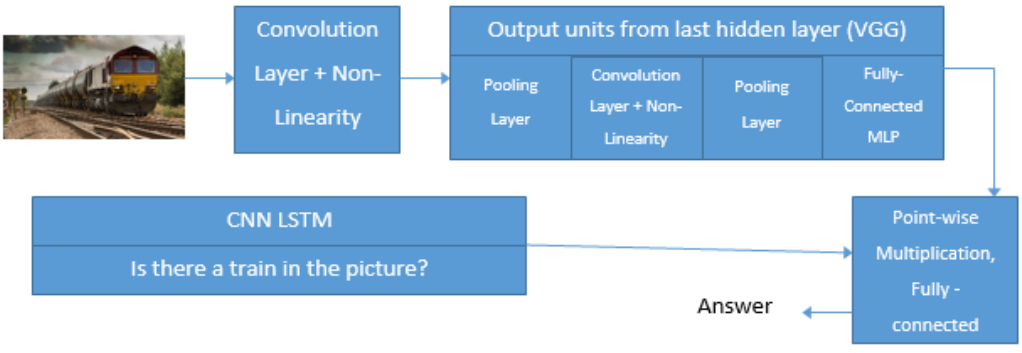

Figure 4. Proposed Long Short-Term Memory (LSTM) based Visual Question Answering Model Integrating Convolutional Layer Non-Linearity.

 doi:10.20944/preprints202505.1872.v1

The annotations embedding is performed by executing the python script as python embedding.py –address embeddings/glove.840B.300d.txt. It is used to train the model for various questions and answers sets. The proposed model is tested using the command python vqa_pranita.py –image file name players.jpg –question "How many players in the picture?". The proposed model is compared with the existing approaches based on the validation error. The comparative analysis is shown in Table 1.

Table 1. Comparison with Existing Approaches.

Approach	Top-1 validation error (%)	Top-5 validation error (%)	Top-5 test error (%)
Proposed Approach (2 nets/ 1 nets, multi-crop, dense eval)	22.8	7.9	7.8
Krizhevsky et al. 2012 (5 nets)	38.1	16.4	16.4
OverFeat Sermanet et al. 2014 (7 nets)	34	13.2	13.6
MSRA He et al., 2014 (1 net)	27.9	9.1	9.1
GooLeNet Szegedy et al., 2014 (7 nets)	-	-	6.7

3. Adaptive Memory Network with Attention Mechanism

The proposed model operates on the input of an image paired with a corresponding question, yielding an output answer to the query. Its design incorporates a convolutional neural network to extract visual features from a variety of sources, including images, videos, or documents. These extracted features are seamlessly embedded within the model using a bi-directional attention-based recurrent neural network. In this framework, attention mechanisms play a pivotal role, with the proposed model offering the flexibility to utilize either GRU (gated recurrent unit) or soft attention models. The GRU-based model excels in reflecting prior context to predict current points, making it well-suited for multistep-ahead predictions. Its simplicity and ease of implementation render it a popular choice. Alternatively, soft attention calculates the context vector by weighting the sum of hidden states from the extracted features. To encode the question, the proposed model utilizes either a GRU recurrent neural network or a positional encoding scheme, ensuring robust information processing. It then employs an adaptive memory network enhanced with an attention mechanism to generate the answer, integrating this amalgamated information effectively. The model's training, evaluation, and testing phases are governed by a set of specified parameters, each tailored to optimize performance within the proposed mechanism. These parameters ensure the model's efficacy across various tasks and datasets, facilitating comprehensive experimentation and analysis.

3.1. Model Architecture

In this paper, the CNN Model options include 'vgg16' and 'resnet50'. VGG-16, a convolutional neural network boasting 16 layers, offers a pre-trained version ('vgg16') trained on a vast ImageNet database, enabling classification across 1000 object categories, spanning keyboards, mice, pencils, and various animals. Typically, CNNs consist of convolution layers, pooling layers, activation layers, and more, with VGG serving as a specific CNN tailored for classification and localization tasks. VGGNet-16, with its uniform architecture comprising 16 convolutional layers, stands out for its reliability and consistency. Like AlexNet, it utilizes 3x3 convolutions with numerous filters, making it a favored choice for feature extraction from images. Although training VGGNet-16 on 4 GPUs requires 2–3 weeks, its publicly available weight configuration serves as a baseline feature extractor across various applications and challenges.

 doi:10.20944/preprints202505.1872.v1

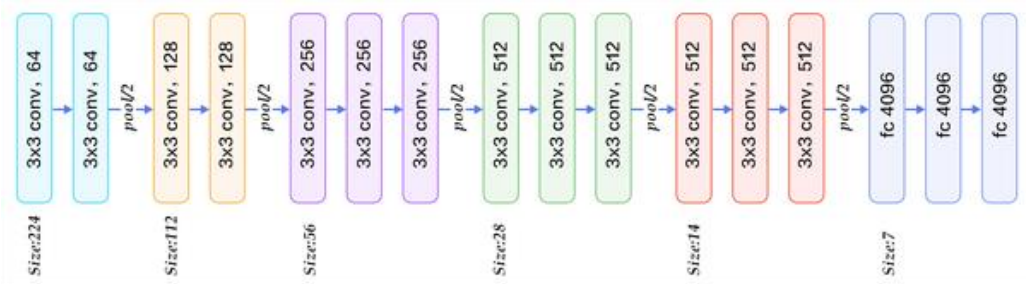

Figure 5. ResNet Architecture used in Proposed System.

However, managing VGGNet's 138 million parameters can pose challenges, which can be mitigated through transfer learning. This involves pre-training the model on a dataset, updating parameters for enhanced accuracy, and leveraging these parameter values. The architecture of VGG16 comprises 16 layers, including convolution layers with varying numbers of filters, max pooling, and fully connected layers, culminating in an output layer with softmax activation. ResNet, short for Residual Networks, represents a classic neural network serving as a backbone for numerous computer vision tasks. ResNet-50, with its 50 layers, offers a pre-trained variant ('resnet50') trained on ImageNet, similarly capable of classifying images across 1000 object categories. The breakthrough of ResNet lies in its ability to train extremely deep neural networks, with variants like Resnet50 denoting the number of layers within the network architecture.

ResNet, an influential neural network introduced by Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun in their 2015 computer vision breakthrough, has made a lasting impact in the field. Its ensemble notably achieved top honors at the ILSVRC 2015 classification competition with an impressive 3.57% error rate. Beyond classification, ResNet excelled in tasks such as ImageNet detection, localization, COCO detection, and COCO segmentation. Among its notable variants, ResNet50 stands out with 48 convolution layers, supplemented by 1 MaxPool and 1 Average Pool layer, collectively performing 3.8 x 10^9 Floating Point Operations. Exploring the 'resnet50' architecture reveals a meticulously structured sequence of operations.

- A convolution with a 7 * 7 kernel, employing 64 different kernels with a stride of size 2.
- Subsequent max pooling with a stride size of 2.
- Following convolutions include kernels of sizes 1 * 1, 3 * 3, and 1 * 1, each repeated 3 times, adding up to 9 layers.
- Further convolutions involve kernels of sizes 1 * 1, 3 * 3, and 1 * 1, repeated 4 times, totaling 12 layers.
- Continuing, kernels of sizes 1 * 1, 3 * 3, and 1 * 1 are employed, iterated 6 times for a sum of 18 layers.
- Additionally, kernels of sizes 1 * 1, 3 * 3, and 1 * 1 are utilized, repeated thrice, resulting in 9 layers.
- The sequence culminates with an average pool operation followed by a fully connected layer featuring 1000 nodes, culminating with a softmax function, contributing 1 layer to the architecture.

The architecture comprises various components, with activation functions and pooling layers excluded from the layer count. Summing up, we have 1 + 9 + 12 + 18 + 9 + 1 = 50 layers, constituting a Deep CNN. Additionally, specific parameters govern the model's behavior. The Maximum Question Length, set at 30, dictates the character limit for questions. The Dimension Embedding is fixed at 512, determining the total text embedding capacity. With 512 GRU Units, the model ensures robust processing capabilities. Gated Recurrent Units (GRU), introduced in 2014 by Kyunghyun Cho et al., play a pivotal role. Similar to LSTM, GRU employs a gating mechanism to regulate memorization. It resembles LSTM but lacks an output gate, resulting in fewer parameters. Moreover, the model operates with a Memory Steps setting of 3, dictating the memory step size permitted within the selected architecture. These parameters collectively contribute to the model's functionality and efficiency in processing textual and sequential data.

To enhance memory updating, the model employs either 'relu' or 'gru'. The Rectifier Linear Unit (ReLU) or GRU serves to introduce non-linearity into images, a crucial aspect given the inherently

 doi:10.20944/preprints202505.1872.v1

non-linear nature of visual data. Images inherently encompass numerous non-linear features, such as pixel transitions, borders, and color variations. Both ReLU and GRU apply fixed functions without the need to retain state information. Furthermore, attention mechanisms, either 'gru' or 'soft', are employed to aid the model in memorizing extensive sequences of data. This is essential as large datasets may contain crucial information that could potentially be overlooked by the model. The GRU-based attention model excels in reflecting previous context to predict subsequent points accurately, making it ideal for multistep-ahead predictions. Its simplicity and ease of implementation further contribute to its widespread adoption. Alternatively, soft attention calculates the context vector by computing a weighted sum of the hidden states of extracted features. This mechanism ensures that relevant information is emphasized during processing, thereby enhancing the model's ability to capture important patterns within the data.

The Tie Memory Weight is set to False. Memory weight tying is a technique where the input-to-hidden and hidden-to-output weights are set to be equal. They are the same object in memory, the same tensor, playing both roles. The hypothesis is that conceptually in a Language Model predicting the next word (converting activations to English words) and converting embeddings to activations are essentially the same operation, the model tasks that are fundamentally similar. It turns out that indeed tying the weights allows a model to train better. Initially it is set to false.

In the realm of Question Encoding, the model provides two pathways: 'gru' or 'positional'. Encoding the question can be achieved through either GRU or positional encoding. The arrangement and sequence of words are pivotal in language, defining both grammar and semantics. Recurrent Neural Networks (RNNs) inherently account for word order during sequential processing. Transformer architecture opts for a multi-head self-attention mechanism over RNN's recurrence, significantly speeding up training and theoretically capturing longer dependencies.

However, without RNN's inherent sequential parsing, Transformers lack explicit word position awareness. To address this, positional encoding is introduced. One approach involves assigning each word a number within the [0, 1] range, representing its position in the sentence. Yet, this approach fails to consistently denote word count within a range across different sentences.

Alternatively, linearly assigning numbers to each time-step (word) faces challenges with potentially large values and inadequate exposure to longer sentences during training. To fulfill key criteria—uniqueness, consistent distance between time-steps, generalization to longer sentences, bounded values, and determinism—positional encoding must strike a delicate balance. The Embed Fact parameter, when set to False, denotes that the model processes input data into ordered vectors termed facts, offering additional information for subsequent stages. This structured approach enhances the model's understanding of the context surrounding the queried question.

3.2. Weight Initialization, Regularization, and Optimization

Initializers play a pivotal role in setting the initial random weights of model layers. In deep neural networks (DNNs), the weights for convolutional and fully connected layers are initialized methodically. They are drawn from a normal distribution with zero mean and a standard deviation determined by the filter kernel dimensions. This ensures that the output variance of each layer remains bounded, preventing issues such as vanishing or exploding gradients. The model's initialization and regularization weights are determined by parameters outlined in Table 2. Additionally, optimization parameters, as shown in Table 3, are crucial for effective model training.

The number of epochs signifies the maximum iterations allowed during training, while the batch size determines the number of input samples processed simultaneously. Among optimization algorithms, Adam stands out for its adaptiveness, often outperforming classical stochastic gradient descent, particularly with sparse datasets. RMSprop, akin to gradient descent with momentum, curbs oscillations in the vertical direction, allowing for larger steps in the horizontal direction, thus facilitating faster convergence. Momentum, or SGD with momentum, expedites gradient vector acceleration, leading to quicker convergence and is widely used in state-of-the-art models.

 doi:10.20944/preprints202505.1872.v1

Table 2. Initialization and Regularization Weights.

Parameters	Set Values
Fully Connected Kernel Initializer Scale	0.08
Fully Connected Kernel Regularizer Scale	1e-6
Fully Connected Activity Regularizer Scale	0.0
Convolution Kernel Regularizer Scale	1e-6
Convolution Activity Regularizer Scale	0.0
Fully Connected Drop Rate	0.5
GRU Drop Rate	0.3

Table 3. Model Optimization Parameters.

Parameters	Set Values
Number of Epochs	100
Batch Size	64
Optimizer	'Adam' or 'RMSProp' or 'Momentum' or 'SGD'
Initial Learning Rate	0.0001
Learning Rate Decay Factor	1.0
Number of Steps per Decay	10000
Clip Gradients	10.0
Momentum	0.0
Use Nesterov	True
Decay	0.9
Centered	True
Beta 1 β1	0.9
Beta 2 β2	0.999
Epsilon	1e-5

The learning rate, a key parameter in optimization algorithms, dictates the step size towards minimizing the loss function at each iteration. Learning rate decay, another crucial technique, involves gradually reducing the learning rate throughout training, promoting both optimization and generalization. Typically, the learning rate is halved every few epochs, following a step decay schedule, which aids in achieving better convergence.

Gradient clipping is a technique that constrains gradient values to a specified range if they exceed certain thresholds, preventing them from becoming too large or too small. This method, along with momentum, is collectively known as "gradient clipping." Momentum enhances the optimization process by incorporating a fraction of the previous weight update into the current one, facilitating faster convergence when gradients persist in the same direction. Nesterov momentum, an enhanced version of momentum, computes the diminishing moving average of gradients at anticipated positions in the search space, rather than at the current positions themselves.

Weight decay, or wd, involves subtracting a constant multiple of the weight from the original weight, in addition to the standard gradient descent step. This technique is employed to regularize the model and prevent overfitting. Mean-subtraction, another preprocessing method, centers the data points by subtracting their mean values, which is particularly useful when dealing with inputs that are predominantly positive or negative.

The hyperparameters β1 and β2 of the optimizer represent the initial decay rates used in estimating the first and second moments of the gradient, which are then exponentially updated at the end of each training step. Epsilon, a crucial parameter, balances exploration and exploitation in optimization algorithms. It determines the probability of choosing exploration over exploitation, allowing the model to make decisions that optimize performance while occasionally exploring alternative options.

 doi:10.20944/preprints202505.1872.v1

The COCO training and validation images can be found at the following link: https://cocodataset.org/#download. After downloading, the COCO training images are stored in the directory: E:\Indivisible\PhD\Pranita Desh\Final Modules\Module 2\train\images, while the COCO validation images are stored in: E:\Indivisible\PhD\Pranita Desh\Final Modules\Module 2\val\images. For the visual question answering (VQA) training and validation data, you can download the questions and annotations from: https://visualqa.org/vqa_v1_download.html. Specifically, store the files mscoco_train2014_annotations.json and OpenEnded_mscoco_train2014_questions.json in the directory: E:\Indivisible\PhD\Pranita Desh\Final Modules\Module 2\train. Similarly, the files mscoco_val2014_annotations.json and OpenEnded_mscoco_val2014_questions.json should be stored in: E:\Indivisible\PhD\Pranita Desh\Final Modules\Module 2\val.

The pre-trained Visual Geometry Group - VGG16 net (vgg16_no_fc.npy) is downloaded from https://app.box.com/s/idt5khauxsamcg3y69jz13w6sc6122ph and Residual Network - ResNet50 net (resnet50_no_fc.npy) is downloaded from https://app.box.com/s/17vthb1zl0zeh340m4gaw0luuf2vscne. These pre-trained model are used to initialize the proposed Memex-CNN RNN VQA (Multimodal Adaptive Memory Convolutional Neural Network and Recurrent Neural Network based Visual Question Answering) model.

The proposed model is developed using Anaconda with Python 3.8 distribution. The python libraries Tensorflow, NumPy, OpenCV, Natural Language Toolkit (NLTK), Pandas, Matplotlib, and tqdm are used to carry implementation of the proposed model. The python MemexCNNRNN.py --phase=train --load_cnn --cnn_model_file= ‘./vgg16_no_fc.npy’ command is executed in Anaconda prompt to train the proposed model. Using this command only RNN part will be trained. To jointly train the CNN and RNN parts python MemexCNNRNN.py --phase=train --load_cnn --cnn_model_file= ‘./vgg16_no_fc.npy’ [--train_cnn] command is executed in Anaconda prompt. The models and checkpoints will be stored in the folder E:\Indivisible\PhD\Pranita Desh\Final Modules\Module 2\models.

To resume training from a checkpoint, run the following command in Anaconda prompt: python MemexCNNRNN.py --phase=train --load --model_file='./models/nameofcheckpoint.npy' [--train_cnn]. To monitor the training progress, execute the command tensorboard --logdir='./summary/' in Anaconda prompt. For evaluating a trained model using validation data, input the command python MemexCNNRNN.py --phase=eval --model_file='./models/MyMemexModel.npy' in Anaconda prompt. To answer questions about images, documents, or videos, ensure that the multimedia data is located in the directory: E:\Indivisible\PhD\Pranita Desh\Final Modules\Module 2\test\images. Prepare a CSV file containing questions, with three fields: image, question, and question_id (refer to Figure 6). Save this CSV file in the directory: E:\Indivisible\PhD\Pranita Desh\Final Modules\Module 2\test.

	A	B	C
1	image	question_id	question
2	landscape.jpg	1	are there mountains in this picture
3	tennis.jpg	2	what game is this man playing
4	horses.jpg	3	how many horses can be seen

Figure 6. CSV File with Three Fields: image, question, question_id.

To get the answers to the questions python MemexCNNRNN.py --phase=test --model_file=‘./models/MyMemexModel.npy’ command is executed in Anaconda prompt. The generated answers are stored in the folder E:\Indivisible\PhD\Pranita Desh\Final Modules\Module 2\test\results. Screenshot shown in Figure 7 indicates the input images stored in folder E:\Indivisible\PhD\Pranita Desh\Final Modules\Module 2\test\images. The generated

 doi:10.20944/preprints202505.1872.v1

results, stored in E:\Indivisible\PhD\Pranita Desh\Final Modules\Module 2\test\results, for the input images and the questions stored in CSV file are shown in Figure 8 screenshot.

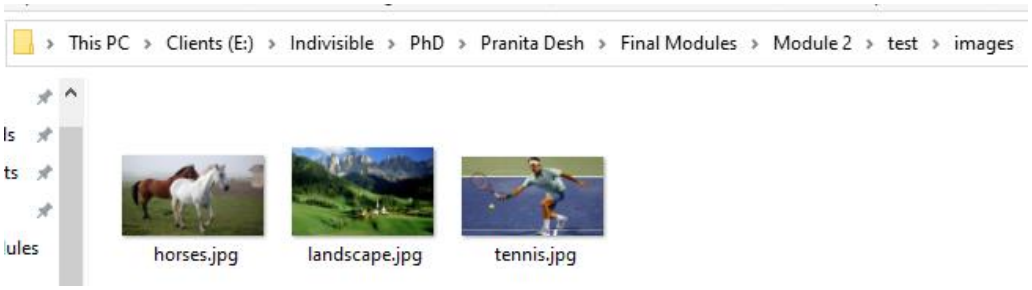

Figure 7. CSV File with Three Fields: image, question, question_id.

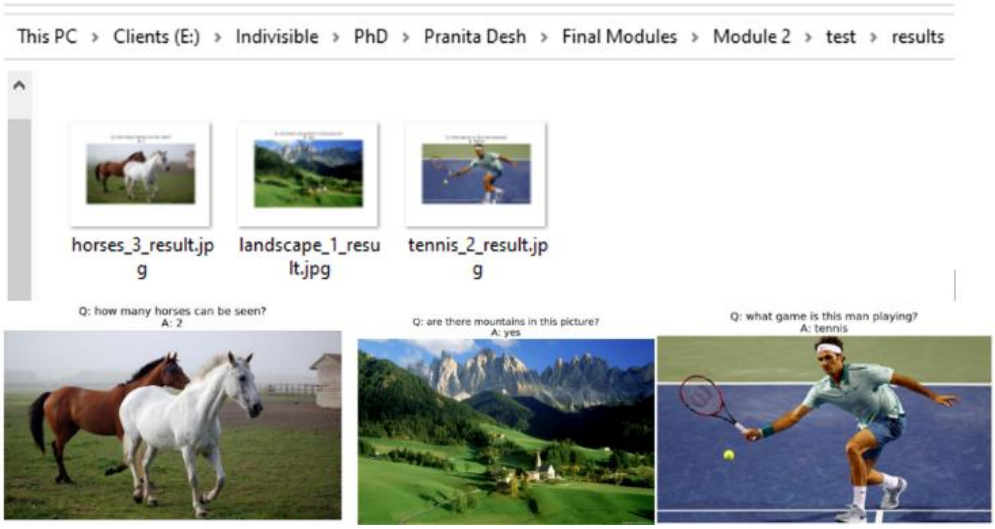

Figure 8. Generated Results.

4. Event-Based Focal Visual-Content Text Attention

In the realm of academic events like STTPs, FDPs, and more, organizers often find themselves inundated with a plethora of photographs. Over the years, academic institutions accumulate thousands of images and hours of video footage, documenting past events ranging from training programs to social gatherings, conferences, and workshops. These visual records serve as invaluable repositories of information, sought after for various purposes. Naturally, one of the most intuitive methods to retrieve information from these archives is by posing questions, such as querying the number of attendees or the date of the event.

To address this formidable challenge, a novel VQA framework called EFVCTA (Event-Focused Visual Content-based Question Answering) has been proposed. This framework is tailored specifically for answering questions related to events captured in photographs by academic institutions or organizations. A system has been developed based on the EFVCTA framework, aimed at facilitating the search for information about past events. Given a sequence of photos depicting an event, the system's objective is to automatically respond to questions regarding the captured events. The input to the EFVCTA system comprises a question and a photograph, while the output consists of a textual answer and a set of evidence photos corroborating the system's generated answer. These evidence images not only aid in validating the response but also furnish additional insights into the queried event.

Despite the proliferation of VQA datasets [6–20] in recent years, none of them cater specifically to the design requirements of the EFVCTA system due to fundamental disparities in the problem

 doi:10.20944/preprints202505.1872.v1

formulation. Unlike conventional datasets, where inputs typically consist of individual images or videos, EFVCTA mandates event photos as input. While some existing datasets may contain image sequences, these are often limited to a single theme or video, rather than encompassing the spectrum of events organized by academic institutions. However, addressing questions in the EFVCTA context necessitates reasoning over multiple photos spanning various events, a capability not adequately supported by existing approaches.

A crucial aspect of the EFVCTA output is the provision of evidence photos, crucial for validating the answers provided. Regrettably, most existing approaches prioritize answer accuracy but fail to furnish such evidence photos to substantiate the generated responses. EFVCTA operates as a multimodal QA framework, leveraging the rich metadata associated with event photos, including title, time, date, and location. Certain questions require a holistic understanding, necessitating the joint consideration of event photos and their associated information. For instance, answering a query like "where was the entrepreneurship camp conducted?" demands synthesizing information from the "entrepreneurship camp" images along with the location metadata. Existing VQA methodologies and datasets often overlook the need for such multimodal information integration.

This work prepares the dataset for designing EFVCTA based system that contains photos of events conducted by institutions and questions about the conducted events. More than 15K questions and answers are constructed on approximately 9K photos of various events or programmes conducted by various institutions. Among 9K images of events 3K images are necessary for answering the questions and for verifying the generated answers. These collected images comprises a wide variety of events such as workshops, conferences, annual social gatherings, training programmes, industrial visits. The annotations assigned to the images based on the question asked to the organizers about the event in all images. The question is supposed to be useful in identifying the information associated with events. An answer comprises the contents based answer as well as some evidence images for justifying the answer.

EFVCTA presents an intriguing and formidable framework. Tackling an EFVCTA question is no simple feat. As a multimodal artificial intelligence construct, EFVCTA holds promise for captivating real-world applications, particularly in navigating the vast and ever-expanding repositories of images and videos housed by diverse institutions. There are few challenges in this new framework. Rich information comprised in the images of events. An institutions or organizations has huge multimedia repositories as images or videos with associated information. For each image or video, the organizers may associate content based annotations, and other metadata. Such input is referred as visual content and the context of the images. A robust method is essential to manage the inconsistently available multimodal content in image annotations. In addition to direct answer useful evidence justifications should get generated. Supporting evidence should get identified for the answers to help organizers of the events to manage images and videos of event.

When asked, "Where was the entrepreneurship camp conducted?" a proficient VQA system should provide a clear answer (e.g., March 12, 2020) along with supporting images to substantiate the answer generation process from the organizer's perspective. For single image the verification process is simple but to examine every image and the generated answer the verification process can take a huge amount of time. So images generated as evidence can be useful to significantly reduce the time required verification of the generated answer. To tackle these challenges, we introduce the Event-based Focal Visual-Content Text Attention (EFVCTA) system, which emphasizes content-based identification. The events are identified on the basis of the video or image contents. Initially the video or images contents are analyzed to identify the similarity between the question’s context and the content of the input video or image. The temporal focal correlation among the question context and input video or image contents are established. This temporal focal correlation is utilized for computing visual-content text attention. The proposed EFVCTA system constructs event-based answers, allowing users to validate the generated responses by providing relevant evidence. This system generates answers to questions by retrieving photos or videos associated with the queried event. The key contributions of this work are:

 doi:10.20944/preprints202505.1872.v1

- Introduce a new multimodal EFVCTA framework that utilizes questions about events organized by institutions or organizations.
- A novel EFVCTA system is proposed to identify the similarity between the question's context and the content of the input video or image which can be used to localize evidence images to justify the generated answers.
- The proposed framework is evaluated by comparing it with the DMN+ (Improved Dynamic Memory Networks) [21], MCB (Multimodal Compact Bilinear Pooling) [22], Soft Attention [23], Soft Attention Bi-directional (BiDAF- Bi-Directional Attention Flow) [24], Spatio-Temporal Reasoning using TGIF Attention [25], and Focal Visual Text Attention (FVTA) [26].

The Event-based Focal Visual-Content Text Attention (EFVCTA) focuses on the content based identification. The events are identified on the basis of the video or image contents. Initially the video or images contents are analyzed to identify the similarity between the question's context and the content of the input video or image. The temporal focal correlation among the question context and input video or image contents are established. This temporal focal correlation is utilized for computing visual-content text attention. The EFVCTA system constructs answers based on event-related photos or videos and generates evidence to support these answers, enabling users to verify their accuracy. This event-focused approach ensures that the responses are grounded in relevant visual content. The entire EFVCTA framework and its methodology are illustrated in Figure 9.

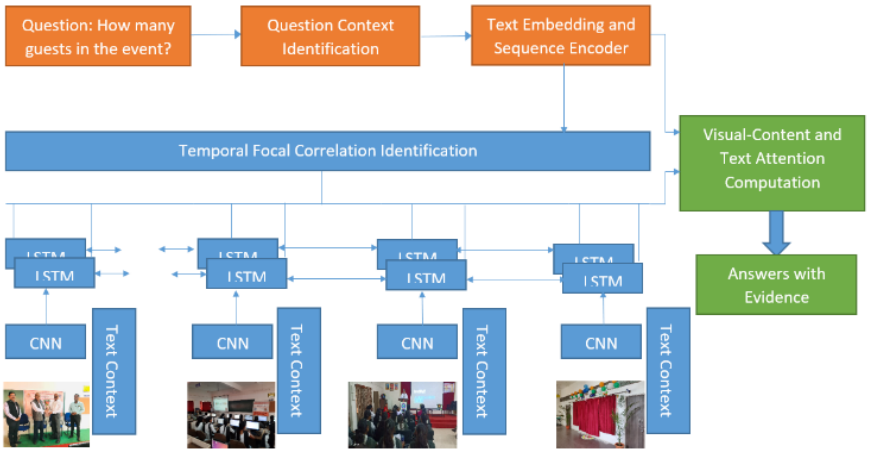

Figure 9. Proposed EFVCTA.

In the proposed EFVCTA framework based system the question context is identified by taking the input image/ the video frame into consideration. The question related to current input image comprises N words. The question is denoted as $Q = w_1, w_2, ..., w_N$. The question context is identified through the input image by identifying the association between the input question words and the images. The question context associated with M images is represented as $Q_{context} = \sum_{i=1}^{M}\sum_{j=1}^{N} I^i{}_{w_j}$. The images I^i for which has the highest $Q_{context}$ will be associated with current question. Those images and the attributes assigned to those images are considered to answer the current question and to generate the evidence related to current question.

The visual content and word (text) attention allows the model to choose the relevant visual contents and question (text) context. The most intermediate answers generated for visual content and text context sequence inputs has temporal duration, localized representation based on visual contents called focal visual-content text context representation. A temporal similarity matrix M is computed. Each entry in similarity matrix m_{ij} represents the similarity between the extracted features of image (visual content) and the question context (word context).

 doi:10.20944/preprints202505.1872.v1

$$m_{ij} = \tanh \sum_{k=1}^{N} param((I_{content}, Q_{context}) + w_{1:N})$$

where *param* are the learning parameters.

The attention mechanism used to compute the visual contents and question context is detailed here. This mechanism applies attention over localized visual contents and their relationship with the question context to identify the most relevant information for answering. The attention score is derived from a tensor T, which represents the interaction between each word in the question and the visual contents of the image. A kernel tensor T is computed between the words of the input question and the localized visual contents $I_{content}$. Each entry in the kernel T_{nm} models the similarity between the n^{th} word in the question and m^{th} image feature.

The implementation of the proposed approach is carried out in Anaconda with Python 3.8 distribution. Initially the MemexQA dataset v1.1 is downloaded from https://memexqa.cs.cmu.edu/memexqa_dataset_v1.1/. The downloaded dataset is preprocessed. The downloaded dataset is modified to include the photos and QA set for various events like conferences, workshops, training programmes, industrial visits, annual social gatherings, FDP, STTP, etc. The training takes carried out on Nvidia GeForce RTX 3080 GPU using about 16 GB GPU memory. The proposed EFVCTA is compared with DMN+ (Improved Dynamic Memory Networks), MCB (Multimodal Compact Bilinear Pooling), Soft Attention, Soft Attention Bi-directional (BiDAF- Bi-Directional Attention Flow), Spatio-Temporal Reasoning using TGIF Attention, and Focal Visual Text Attention (FVTA). The test accuracies of these models compared with the proposed EFVCTA is shown in Table 4. The generated answers to the asked questions and corresponding evidences are shown in Figure 10.

Table 4. Comparative Analysis of Various Models.

Model	Test Accuracy
DMN+ (Improved Dynamic Memory Networks)	48.51%
MCB (Multimodal Compact Bilinear Pooling)	46.23%
Soft Attention	62.08%
Soft Attention Bi-directional (BiDAF- Bi-Directional Attention Flow)	60.09%
Spatio-Temporal Reasoning using TGIF Attention	63.06%
Focal Visual Text Attention (FVTA)	66.86%
Proposed Event-based Focal Visual-Content Text Attention (EFVCTA)	68.07%
Model	Test Accuracy

Figure 10. Screenshot of generated results with the evidence images for justification.

 doi:10.20944/preprints202505.1872.v1

5. Comprehensive Analysis and Impact of Proposed Event-Based Local Focal Visual-Content Text Attention

The Proposed EFVCTA model outperforms all baselines with the highest test accuracy of 68.07%. Traditional attention mechanisms like Soft Attention and BiDAF perform moderately well (62.08%, 60.09%). Models with visual-text fusion mechanisms (FVTA and EFVCTA) tend to perform better, suggesting the importance of aligning visual and textual information. Models like DMN+ and MCB, while important historically, show relatively lower performance in this specific task domain. The proposed EFVCTA model leverages event-based modeling and local focal mechanisms, which likely enable it to better capture relevant contextual cues and temporal dependencies in both visual and textual modalities. This might explain its superior performance in past event search tasks, where both precise timing and multimodal understanding are crucial.

The Event-Based Local Focal Visual-Content Text Attention (EFVCTA) model demonstrates a significant leap in performance for past event search tasks, achieving a test accuracy of 68.07%, the highest among all compared models. EFVCTA surpasses traditional models like DMN+ (48.51%) and MCB (46.23%) by a margin of nearly 20%, highlighting the limitations of generic memory and pooling techniques in understanding complex temporal-visual interactions. Compared to advanced attention-based models such as FVTA (66.86%) and TGIF Attention (63.06%), EFVCTA still maintains a noticeable edge, showcasing the strength of its event-centric and local focal mechanisms. The focused alignment of visual and textual cues around events, combined with temporal localization, makes EFVCTA particularly effective in retrieving and understanding event-specific content from multimodal inputs. Event-Based Modeling: Enhances temporal relevance by anchoring attention mechanisms around detected events rather than treating the entire sequence uniformly.

Local focal attention narrows attention to the most informative regions, reducing noise and improving precision in visual-text alignment. Multimodal synergy with stronger integration of visual and textual streams enables more coherent interpretation of context-rich scenarios. The proposed EFVCTA model represents a notable advancement in the field of event-based retrieval and reasoning. By effectively combining temporal, visual, and textual information with a localized attention mechanism, EFVCTA sets a new benchmark for intelligent past event search systems, paving the way for smarter video understanding and memory-based AI applications.

Table 5. Comprehensive Analysis of Proposed EFVCTA.

Model	Visual-Text Alignment	Temporal Awareness	Attention Mechanism	Multimodal Fusion	Complexity
DMN+ (Improved Dynamic Memory Networks)	Poor	Moderate (via memory)	Soft Attention	No	Low
MCB (Multimodal Compact Bilinear Pooling)	Moderate	Absent	None	Bilinear Pooling	Moderate
Soft Attention	Moderate	Absent	Soft Attention	No	Low
Soft Attention Bi-directional (BiDAF-Bi-Directional Attention Flow)	Strong (textual)	Implicit	Bi-directional Flow	No	Moderate
Spatio-Temporal Reasoning using TGIF Attention	Moderate	Strong (TGIF events)	Spatio-Temporal Attention	Early Fusion	High
Focal Visual Text Attention (FVTA)	Strong	Strong	Focal Attention	Late Fusion	High
Proposed Event-based Focal Visual-Content Text Attention (EFVCTA)	Excellent (event-localized)	Excellent (event-based focal)	Local Focal Attention	Hierarchical Fusion	

The more comprehensive analysis is shown in table V. In table visual-text alignment shows how effectively the model aligns visual and textual modalities. Temporal awareness indicates the model's ability to handle time-based dependencies. Attention mechanism shows the sophistication of its

 doi:10.20944/preprints202505.1872.v1

attention strategy. Multimodal fusion shows how the model integrates visual and textual data (e.g., early, late, or hierarchical fusion). While complexity indicates an estimate of model complexity based on architecture and computation requirements. DMN+ and MCB struggle with aligning visual and textual data effectively due to the absence of specialized mechanisms for modality interaction. Models like Soft Attention, BiDAF, and TGIF Attention show moderate alignment capabilities. FVTA improves this significantly using focal attention, but the Proposed EFVCTA leads with event-localized alignment, which ensures attention is focused on relevant visual regions that match the event semantics in text.

MCB and Soft Attention lack temporal modeling, making them less effective for sequences where timing is crucial. DMN+, BiDAF, and TGIF Attention introduce varying degrees of temporal handling, through memory structures or spatio-temporal attention. FVTA and EFVCTA excel with strong, explicit temporal modeling, with EFVCTA further refining this by focusing on event-specific time spans, improving relevance and context understanding. Basic attention mechanisms like soft attention are employed by earlier models (DMN+, Soft Attention). BiDAF improves upon this with bi-directional attention flows, better capturing contextual relationships. TGIF Attention and FVTA integrate temporal and focal components into their attention mechanisms. EFVCTA introduces a Local Focal Attention strategy, which not only concentrates on event-relevant regions but also filters out irrelevant information, making attention more precise and context-aware.

DMN+ and Soft Attention don't offer strong multimodal fusion—text and visuals are processed largely independently. MCB uses bilinear pooling to combine modalities, which is effective but computationally expensive. TGIF Attention and FVTA apply structured fusion (early/late), improving integration. The Proposed EFVCTA utilizes hierarchical fusion, allowing it to combine text and visual data at multiple levels of granularity, enhancing semantic alignment. Simpler models like DMN+, MCB, and Soft Attention are lightweight and easier to train, but their simplicity limits their understanding of complex, multimodal event structures. BiDAF, TGIF Attention, and FVTA add complexity for better modeling of temporal and multimodal data. EFVCTA, with its advanced attention and fusion techniques, is more complex, but this added sophistication allows it to handle nuanced, context-rich past event search tasks much more effectively.

6. Practical Applications and Potential Limitations

EFVCTA can efficiently pinpoint the exact frames where a relevant event occurred by aligning visual actions with descriptive queries, saving time and improving situational awareness. Event-localized attention allows accurate matching between natural language queries and visual content, ideal for newsrooms, documentaries, and entertainment. EFVCTA understands both when and what happened, making it perfect for time-sensitive question answering in videos. Its ability to focus attention on both the textual description and corresponding visual segments enables precise and meaningful highlight generation. Event-based modeling ensures critical incidents are detected with temporal accuracy and low false alarms. EFVCTA can help legal professionals navigate through large volumes of video data to find and present event-relevant evidence efficiently. EFVCTA can retrieve exact moments in recorded lectures where certain concepts were discussed, aiding in personalized learning.

The EFVCTA model's hierarchical fusion and focal attention mechanisms introduce additional layers and operations. It requires more memory and compute resources, which can limit real-time performance or deployment on edge devices. EFVCTA relies heavily on accurate event segmentation or detection to guide attention. If events are misidentified or poorly defined, the model may attend to irrelevant segments, degrading performance. The model is trained and tuned for specific types of event-based video and query datasets. Transferring to domains like medical imaging, industrial monitoring, or cartoon/video game analysis may require retraining and domain-specific adaptation. Natural language queries can be vague, subjective, or context-dependent (e.g., "when the mood changed"). Without external knowledge or sentiment modeling, the model may struggle to interpret such queries effectively. The model benefits from detailed multimodal training data with aligned visual segments and text. Collecting and annotating such data is expensive and time-consuming,

 doi:10.20944/preprints202505.1872.v1

especially for custom or underrepresented domains. In high-stakes applications (e.g., legal, healthcare), the lack of interpretability can be a barrier to trust and adoption. Event localization and attention computation introduce latency which limits the ability to scale EFVCTA in real-time systems like live surveillance or autonomous navigation. While EFVCTA introduces valuable innovations for event-based multimodal search, its efficiency, adaptability, and robustness must be further optimized for broader, real-world deployment.

7. Conclusions

This research tackles the challenge of identifying event-specific photos or videos from a vast collection that includes workshops, Short Term Training Programs (STTPs), Faculty Development Programs (FDPs), conferences, annual social gatherings, and more. It involves analyzing various visual question-answering methods for event search. A dataset is curated and utilized for searching past events, and a question understanding network model is designed and developed. Additionally, a content-based and text-based event search engine for photos or videos is created as part of this research. A text-based answer generation engine has been developed and integrated to produce photo or video evidence corresponding to the generated answers. The performance of the proposed approaches is thoroughly evaluated.

The research objectives are achieved through the proposal, design, and development of various models, as detailed below.

- In this work LSTM based VQA model which integrates convolutional layer non-linearity is proposed for addressing question understanding
- An adaptive memory network is developed with attention mechanism which utilizes weight regularization and optimization for addressing content based and text based photos or videos search
- A novel strategy event-based focal visual-content text attention (EFVCTA) model is proposed, designed and developed for question context identification, text embedding with sequence encoder, temporal focal correlation identification, and visual content-text attention for past event search with evidence generation

The proposed methodologies are evaluated using standard datasets. The dataset for workshops, Short Term Training Programmes (STTP's), Faculty Development Programmes (FDP's), conferences, annual social gatherings, etc. is created for evaluating proposed EFVCTA model. Various parameters are utilized for the evaluation proposed models. The proposed models have the potential to be extended for identifying war crimes, human rights violations, and terrorist activities.

Author Contributions: Conceptualization, Pranita P. Deshmukh, and S. Poonkuntran; methodology, Pranita P. Deshmukh, and S. Poonkuntran; software, Pranita P. Deshmukh, and S. Poonkuntran; validation, Pranita P. Deshmukh, and S. Poonkuntran; formal analysis, Pranita P. Deshmukh, and S. Poonkuntran; investigation, Pranita P. Deshmukh, and S. Poonkuntran; writing—original draft preparation, Pranita P. Deshmukh, and S. Poonkuntran; writing—review and editing, Pranita P. Deshmukh, and S. Poonkuntran; visualization, Pranita P. Deshmukh, and S. Poonkuntran; supervision, S. Poonkuntran. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

EFVCTA	Event-based Focal Visual-Content Text Attention
LSTM	Long Short-Term Memory
VQA	Visual Question Answering
CNN	Convolutional Neural Network
DMN+	Improved Dynamic Memory Networks

 doi:10.20944/preprints202505.1872.v1

MCB	Multimodal Compact Bilinear Pooling
BiDAF	Bi-Directional Attention Flow
FVTA	Focal Visual Text Attention
STTPs	Short Term Training Programs
FDPs	Faculty Development Programs

References

1. Sepp Hochreiter; Jürgen Schmidhuber (1997). "Long short-term memory". 9 (8): 1735–1780. Doi: 10.1162/neco.1997.9.8.1735.
2. Graves, A.; Liwicki, M.; Fernandez, S.; Bertolami, R.; Bunke, H.; Schmidhuber, J. (2009). "A NovelConnectionist System for Improved Unconstrained Handwriting Recognition". IEEE Transactions on Pattern Analysis and Machine Intelligence. 31 (5): 855–868.
3. Sak, Hasim; Senior, Andrew; Beaufays, Francoise (2014). "Long Short-Term Memory recurrentneural network architectures for large scale acoustic modeling". Archived from the original (https://static.googleusercontent.com/media/research. google.com/en//pubs/ archive/43905.pdf) on 2021-09-24.
4. Li, Xiangang; Wu, Xihong (2014). "Constructing Long Short-Term Memory based DeepRecurrent Neural Networks for Large Vocabulary Speech Recognition". arXiv:1410.4281 (https://arxiv.org/abs/1410.4281)
5. Calin, Ovidiu. Deep Learning Architectures. Cham, Switzerland: Springer Nature. p. 555. ISBN 978-3-030-36720-6.
6. H. Yang, L. Chaisorn, Y. Zhao, S.-Y. Neo, and T.-S. Chua, "VideoQA: Question answering on news video," in Proc. 11th ACM Int. Conf. Multimedia, 2003, pp. 632–641.
7. S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. Lawrence Zitnick, and D. Parikh, "VQA: Visual question answering," in Proc. IEEE Int. Conf. Comput. Vis., 2015, pp. 2425–2433.
8. Y. Zhu, O. Groth, M. Bernstein, and L. Fei-Fei, "Visual7w: Grounded question answering in images," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 4995–5004.
9. Y. Jang, Y. Song, Y. Yu, Y. Kim, and G. Kim, "TGIF-QA: Toward spatio-temporal reasoning in visual question answering," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 1359–1367.
10. M. Tapaswi, Y. Zhu, R. Stiefelhagen, A. Torralba, R. Urtasun, and S. Fidler, "MovieQA: Understanding stories in movies through question-answering," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 4631–4640.
11. H. Xu and K. Saenko, "Ask, attend and answer: Exploring question-guided spatial attention for visual question answering," in Proc. Eur. Conf. Comput. Vis., 2016, pp. 451–466.
12. H. Gao, J. Mao, J. Zhou, Z. Huang, L. Wang, and W. Xu, "Are you talking to a machine? Dataset and methods for multilingual image question," in Proc. 28th Int. Conf. Neural Inf. Process. Syst., 2015, pp. 2296–2304.
13. J. Andreas, M. Rohrbach, T. Darrell, and D. Klein, "Neural module networks," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 39–48.
14. J. Johnson, B. Hariharan, L. van der Maaten, L. Fei-Fei, C. L. Zitnick, and R. Girshick, "CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 1988–1997.
15. K. Kafle and C. Kanan, "An analysis of visual question answering algorithms," in Proc. IEEE Int. Conf. Comput. Vis., 2017, pp. 1983–1991.
16. L. Zhu, Z. Xu, Y. Yang, and A. G. Hauptmann, "Uncovering temporal context for video question and answering," Int. J. Comput. Vis., vol. 124, no. 3, pp. 409–421, 2017.
17. M. Ren, R. Kiros, and R. Zemel, "Exploring models and data for image question answering," in Proc. 28th Int. Conf. Neural Inf. Process. Syst., 2015, pp. 2953–2961.
18. L. Yu, E. Park, A. C. Berg, and T. L. Berg, "Visual madlibs: Fill in the blank description generation and question answering," in Proc. IEEE Int. Conf. Comput. Vis., 2015, 2461–2469.
19. Y. Goyal, T. Khot, D. Summers-Stay, D. Batra, and D. Parikh, "Making the V in VQA matter: Elevating the role of image understanding in visual question answering," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 6325–6334.

 doi:10.20944/preprints202505.1872.v1

20. D. Xu, Z. Zhao, J. Xiao, F. Wu, H. Zhang, X. He, and Y. Zhuang, "Video question answering via gradually refined attention over appearance and motion," in Proc. 25th ACM Int. Conf. Multimedia, 2017, pp. 1645–1653.
21. Caiming Xiong, Stephen Merity, and Richard Socher, "Dynamic Memory Networks for Visual and Textual Question Answering," arXiv:1603.01417v1 [cs.NE] 4 Mar 2016.
22. Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach, "Multimodal Compact Bilinear Pooling for Visual Question Answering and Visual Grounding," arXiv:1606.01847v3 [cs.CV] 24 Sep 2016.
23. Mateusz Malinowski, Carl Doersch, Adam Santoro, and Peter Battaglia, "Learning Visual Question Answering by Bootstrapping Hard Attention," arXiv:1808.00300v1 [cs.CV] 1 Aug 2018.
24. Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, Hananneh Hajishirzi, "Bi-Directional Attention Flow for Machine Comprehension," arXiv:1611.01603v6 [cs.CL] 21 Jun 2018.
25. Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim, "TGIF-QA: Toward Spatio-Temporal Reasoning in Visual Question Answering," arXiv:1704.04497v3 [cs.CV] 3 Dec 2017.
26. Junwei Liang, Lu Jiang, Liangliang Cao, Yannis Kalantidis, Li-Jia Li, and Alexander G. Hauptmann, "Focal Visual-Text Attention for Memex Question Answering," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 41, No. 8, pp. 1893-1908, Aug 2019.