

IMAGE CAPTION GENERATOR USING CNN AND LSTM

CONTENT
Abstract
1.INTRODUCTION
1.1 Motivation
1.2 Problem Statement
1.3 Objective of the Project
2.LTERATURE SURVEY
2.1 Related Work
3. PROPOSED SYSTEM
3.1 Advantages
3.2 work Flow of Proposed system
4. REQUIREMENT ANALYSIS
4.1 Function and non-functional requirements
4.2 Hardware Requirements
4.3 Software Requirements
4.4 Architecture
5. METHODOLOGY
5.1 Resnet-50
5.2 LSTM
6. UML DIAGRAMS
6.1 UML Diagram(class, use case, sequence, collaborative, deployment, activity, ER diagram and Component diagram)
6.2 Data Flow Diagram

7. IMPLEMENTATION AND RESULTS
7.1 Modules
7.2 Output Screens
10. CONCLUSION
11. FUTURE ENHANCEMENT
12. REFERENCES

ABSTRACT

The provided script implements an image captioning model using the image dataset. The architecture combines a ResNet50 convolutional neural network (CNN) for image feature extraction and a long short-term memory network (LSTM) for processing word sequences. After reading and cleaning captions, the script preprocesses the data, extracts image features using ResNet50, and prepares the training and test datasets. The model is designed to predict captions given an image, and it incorporates word embeddings from GloVe. The script also involves creating word-to-index and index-to-word mappings, defining the model architecture, and training the model using a generator for data loading. The training utilizes a combination of image features and word sequences, and the model is evaluated using BLEU scores on test images. The overall approach reflects a deep learning paradigm for image captioning, leveraging both visual and linguistic information to generate descriptive captions. The ResNet50 CNN serves as a powerful feature extractor, and the LSTM captures sequential dependencies in language, resulting in a comprehensive image captioning model.

KEYWORDS: CNN, Resnet-50, image caption generation, LSTM..

1. INTRODUCTION

1.1 MOTIVATION

The motivation behind this image captioning project stems from a commitment to improving accessibility for individuals with visual impairments and advancing the capabilities of multimodal artificial intelligence. By employing sophisticated deep learning models, such as ResNet-50 for image feature extraction and LSTMs for sequence modeling, the project aims to generate accurate and contextually rich captions for diverse visual content. The envisioned impact extends beyond accessibility to encompass improved human-computer interaction, practical applications in content retrieval and image indexing, and valuable contributions to the broader research community. Leveraging state-of-the-art architectures not only benchmarks the effectiveness of deep learning models but also offers educational value by providing a hands-on example for practitioners and researchers to explore the complexities of deep learning, natural language processing, and multimodal systems. Ultimately, the project's motivation lies in its potential to enhance the understanding and communication between machines and humans in the realm of visual information processing.

1.2 PROBLEM STATEMENT

The primary problem addressed by this project is the inadequacy of existing image captioning systems in producing comprehensive and accurate captions. The existing image captioning systems face challenges in generating accurate and contextually relevant descriptions, often resulting in inadequate or inaccurate captions. The main problem addressed by this project is the development of an innovative system to produce comprehensive and precise image captions by exploiting both global and specific image features.

1.3 OBJECTIVE OF THE PROJECT

The primary objective of this project is to develop a robust and accurate automatic image captioning system using advanced deep learning techniques. This involves implementing a multimodal model that effectively integrates visual features extracted by the ResNet50 convolutional neural network with sequential information processed by long short-term memory networks (LSTMs). The specific objectives include preprocessing and cleaning the caption dataset, creating word-to-index and index-to-

word mappings, obtaining word embeddings from GloVe, and training the model to generate coherent and contextually relevant captions. Evaluation will be conducted using established metrics like BLEU scores to assess the model's performance. Additionally, the project aims to contribute to the field by providing insights into the challenges and advancements in image captioning, showcasing its practical applications, and potentially laying the foundation for further research in multimodal AI systems.

LITERATURE REVIEW

2.1 Related Work:

[1]. M. Sailaja; K. Harika; B. Sridhar; Rajan Singh, Image Caption Generator using Deep Learning: 2022 International Conference on Advancements in Smart, Secure and Intelligent Computing (ASSIC)

Over the last few years deep neural network made image captioning conceivable. Image caption generator provides an appropriate title for an applied input image based on the dataset. The present work proposes a model based on deep learning and utilizes it to generate caption for the input image. The model takes an image as input and frame the sentence related to the given input image by using some algorithms like CNN and LSTM. This CNN model is used to identify the objects that are present in the image and Long Short-Term Memory (LSTM) model will not only generate the sentence but summarize the text and generate the caption that is suitable for the project. So, the proposed model mainly focuses on identify the objects and generating the most appropriate title for the input images.

[2]. C. S. Kanimozhiselvi; Karthika V; Kalaivani S P; Krithika S, Image Captioning Using Deep Learning, 2022 International Conference on Computer Communication and Informatics (ICCCI).

The process of generating a textual description for images is known as image captioning. Now a days it is one of the recent and growing research problem. Day by day various solutions are being introduced for solving the problem. Even though, many solutions are already available, a lot of attention is still required for getting better and precise results. So, we came up with the idea of developing a image captioning model using different combinations of Convolutional Neural Network architecture along with Long Short Term Memory in order to get better results. We have used three combination of CNN and

LSTM for developing the model. The proposed model is trained with three Convolutional Neural Network architecture such as Inception-v3, Xception, ResNet50 for feature extraction from the image and Long ShortTerm Memory for generating the relevant captions. Among the three combinations of CNN and LSTM, the best combination is selected based on the accuracy of the model. The model is trained using the Flickr8k dataset.

[3]. Chetan Amritkar; Vaishali Jabade, Image Caption Generation Using Deep Learning Technique, 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)

In Artificial Intelligence (AI), the contents of an image are generated automatically which involves computer vision and NLP (Natural Language Processing). The neural model which is regenerative, is created. It depends on computer vision and machine translation. This model is used to generate natural sentences which eventually describes the image. This model consists of Convolutional Neural Network (CNN) as well as Recurrent Neural Network (RNN). The CNN is used for feature extraction from image and RNN is used for sentence generation. The model is trained in such a way that if input image is given to model it generates captions which nearly describes the image. The accuracy of model and smoothness or command of language model learns from image descriptions is tested on different datasets. These experiments show that model is frequently giving accurate descriptions for an input image.

[4]. Varsha Kesavan; Vaidehi Muley; Megha Kolhekar: Deep Learning based Automatic Image Caption Generation. 2019 Global Conference for Advancement in Technology (GCAT)

The paper aims at generating automated captions by learning the contents of the image. At present images are annotated with human intervention and it becomes nearly impossible task for huge commercial databases. The image database is given as input to a deep neural network (Convolutional Neural Network (CNN)) encoder for generating "thought vector" which extracts the features and nuances out of our image and RNN (Recurrent Neural Network) decoder is used to translate the features and objects given by our image to obtain sequential, meaningful description of the image. In this paper, we systematically analyze different deep neural network-based image caption generation approaches and pretrained models to conclude on the most efficient model with fine-tuning. The analyzed models contain both with and without 'attention' concept to optimize the caption generating ability of the model. All the models are trained on the same dataset for concrete comparison.

3 PROPOSED METHOD

Our proposed system employs combination of ResNet-50 and LSTM ensures a seamless fusion of visual and linguistic information. The ResNet-50 feature vector serves as a foundation for the LSTM to generate contextually relevant captions, effectively marrying the strengths of both modalities. The proposed architecture aims to overcome challenges associated with understanding complex visual scenes and maintaining linguistic context, ultimately leading to improved image captioning performance. The use of ResNet-50 as a feature extractor and LSTM for sequence modeling represents a state-of-the-art approach in the field, aligning with contemporary advancements in deep learning for multimodal tasks.

3.1 ADVANTAGES:

Rich Visual Representations:

ResNet-50: ResNet-50, a powerful convolutional neural network, excels at capturing rich visual features from images. Its deep architecture allows it to learn hierarchical representations, enabling the extraction of intricate details and patterns.

High-Level Semantic Features:

ResNet-50: ResNet-50 provides high-level semantic features that go beyond simple object detection. This is crucial for generating captions that not only describe objects but also capture the semantic context and relationships within a scene.

Effective Handling of Varied Image Sizes:

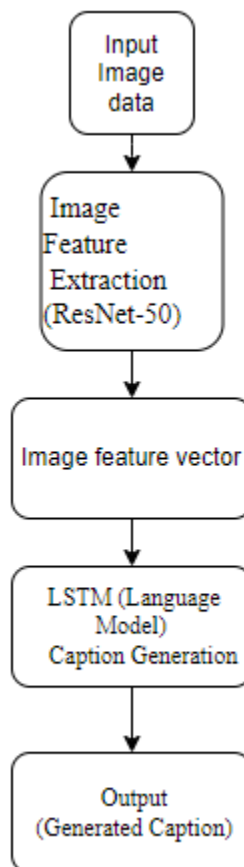
ResNet-50: ResNet-50 can handle images of various sizes without requiring manual resizing. This flexibility is advantageous when working with datasets containing images of different resolutions.

Sequential Context Understanding:

LSTM: LSTMs are well-suited for processing sequential data and understanding long-term dependencies. In image captioning, this allows the model to generate coherent and contextually relevant captions by considering the sequential nature of language.

Contextual Adaptation:

LSTM: LSTMs adapt dynamically to the context of the input sequence, adjusting the weightings on different elements based on their relevance to the current state. This adaptability is crucial for generating captions that evolve meaningfully over time.

3.2 FLOW OF THE PROJECT:

4. REQUIREMENT ANALYSIS

4.1 Functional and non-functional requirements

Requirement's analysis is very critical process that enables the success of a system or software project to be assessed. Requirements are generally split into two types: Functional and non-functional requirements.

Functional Requirements: These are the requirements that the end user specifically demands as basic facilities that the system should offer. All these functionalities need to be necessarily incorporated into the system as a part of the contract. These are represented or stated in the form of input to be given to the system, the operation performed and the output expected. They are basically the requirements stated by the user which one can see directly in the final product, unlike the non-functional requirements.

Examples of functional requirements:

- 1) Authentication of user whenever he/she logs into the system
- 2) System shutdown in case of a cyber-attack
- 3) A verification email is sent to user whenever he/she register for the first time on some software system.

Non-functional requirements: These are basically the quality constraints that the system must satisfy according to the project contract. The priority or extent to which these factors are implemented varies from one project to other. They are also called non-behavioral requirements.

They basically deal with issues like:

- Portability
- Security
- Maintainability
- Reliability
- Scalability
- Performance
- Reusability

- Flexibility

Examples of non-functional requirements:

- 1) Emails should be sent with a latency of no greater than 12 hours from such an activity.
- 2) The processing of each request should be done within 10 seconds
- 3) The site should load in 3 seconds whenever of simultaneous users are > 10000

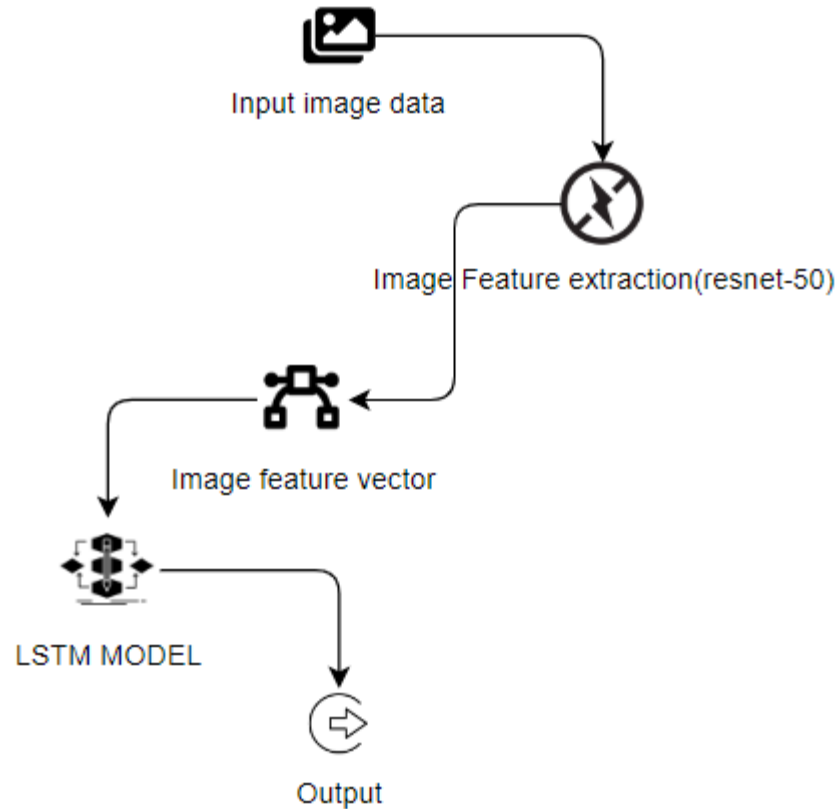
4.2 HARDWARE REQUIREMENTS

Processor	- I3/Intel Processor
Hard Disk	- 160GB
Key Board	- Standard Windows Keyboard
Mouse	- Two or Three Button Mouse
Monitor	- SVGA
RAM	- 8GB

4.3 SOFTWARE REQUIREMENTS:

Operating System	: Windows 7/8/10
Server side Script	: HTML, CSS, Bootstrap & JS
Programming Language	: Python
Libraries	: Flask, Pandas, Mysql.connector, Os, Smtplib, Numpy
IDE/Workbench	: PyCharm
Technology	: Python 3.6+
Server Deployment	: Xampp Server
Database	: MySQL

4.4 ARCHITECTURE:



5. METHODOLOGY

5.1 Resnet-50:

ResNet (Residual Network) is a type of deep neural network architecture designed to address the vanishing gradient problem during training of deep convolutional neural networks (CNNs). ResNet introduces skip connections, also known as residual connections, which allow the network to learn residual functions. These skip connections pass the input directly to the output of deeper layers, enabling the model to skip over certain layers. This helps in mitigating the vanishing gradient problem, making it easier to train very deep networks.

In the context of an image caption generator, ResNet can play a crucial role in feature extraction from images. The encoder part of an image captioning model typically uses a pre-trained CNN, such as ResNet, to extract meaningful features from the input images. The idea is to leverage the knowledge learned by the pre-trained ResNet model on a large dataset (e.g., ImageNet) to capture high-level features in images.

Here's how the ResNet model can be integrated into an image caption generator:

Pre-trained ResNet as Image Encoder:

The ResNet model is used as a feature extractor for images.

The model is typically pre-trained on a large dataset for image classification tasks (e.g., ImageNet).

The weights learned during pre-training capture hierarchical and abstract features in images.

Feature Extraction:

Given an input image, the pre-trained ResNet model is used to extract features from intermediate layers.

The features represent high-level visual information present in the image.

Integration with Captioning Model:

The extracted image features are then passed to the decoder part of the image captioning model.

The decoder, often implemented as a recurrent neural network (RNN) or transformer, generates a textual description of the image based on the input features.

5.2 LSTM:

Long Short-Term Memory (LSTM) is a type of recurrent neural network (RNN) architecture designed to capture and learn long-term dependencies in sequential data. While LSTMs are commonly used for natural language processing tasks, they can also play a crucial role in image caption generators.

Here's how LSTMs are typically incorporated into an image caption generator:

Sequence Modeling:

In the context of image captioning, LSTMs are used to model sequential information, such as generating a sequence of words in a sentence.

The LSTM is employed as the decoder part of the image captioning model, taking as input the features extracted from the image.

Image Feature Input:

The LSTM receives the image features (extracted by a pre-trained CNN like ResNet) as an initial input. These features serve as the context or starting point for generating the image caption.

Word Generation:

The LSTM generates words one at a time, considering the context provided by the image features and the previously generated words.

At each time step, the LSTM produces a probability distribution over the vocabulary, and a word is sampled from this distribution.

Recurrent Connections:

LSTMs have recurrent connections that allow them to maintain and update an internal memory state, which helps capture long-term dependencies in the sequence.

The internal state is updated at each time step based on the input features and the previously generated word.

Training:

During training, the model is optimized to minimize the difference between the predicted caption and the ground truth caption.

The loss is computed based on the generated word probabilities at each time step.

Word Embeddings:

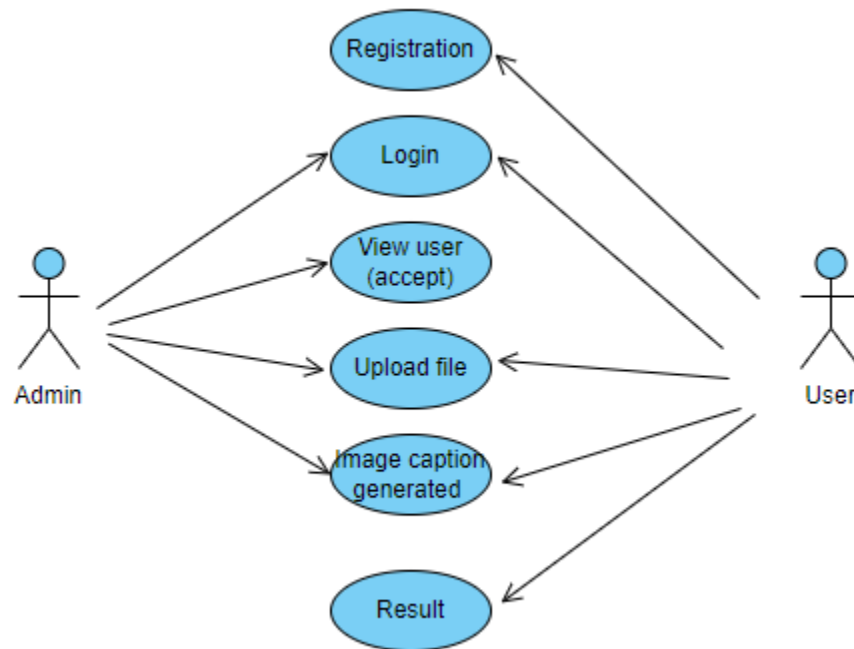
To deal with the discrete nature of words, word embeddings are often used to represent words as continuous vectors.

The LSTM generates these embeddings, which are then used to predict the next word.

By using an LSTM as the decoder in an image caption generator, the model can effectively capture the dependencies between words in a sentence and generate coherent and contextually relevant captions for images. The combination of a pre-trained image encoder (e.g., ResNet) and an LSTM-based decoder allows the model to leverage both visual information from the image and linguistic context to produce meaningful and descriptive captions

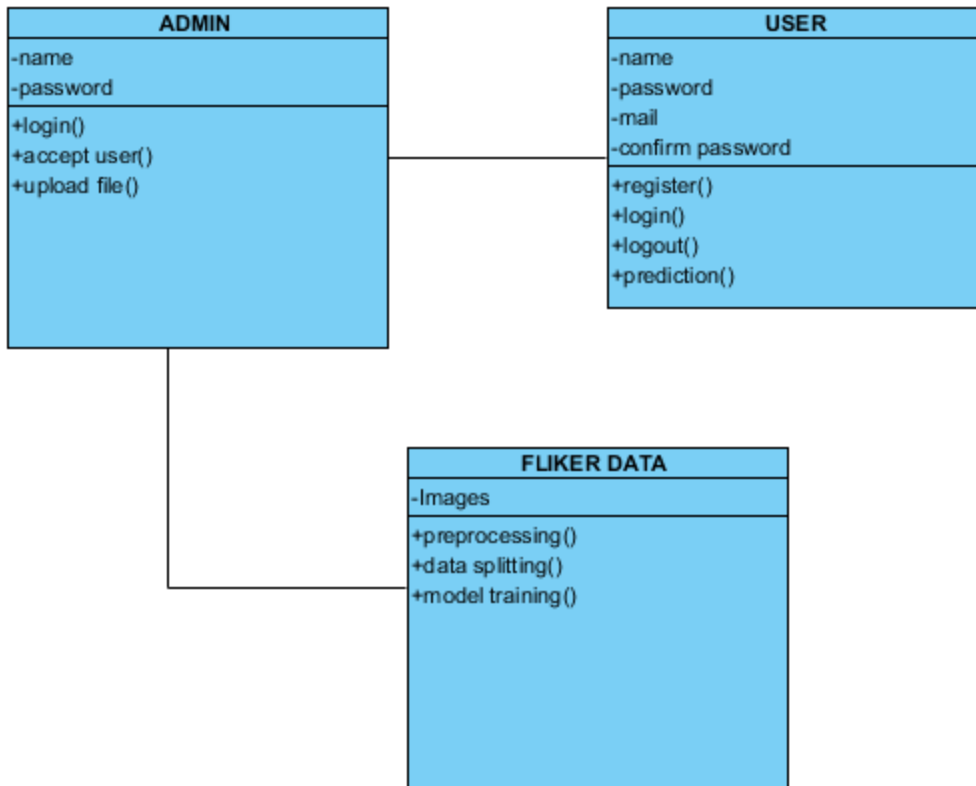
6 UML Diagrams:**6.1 USE CASE DIAGRAM:**

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



6.2 CLASS DIAGRAM:

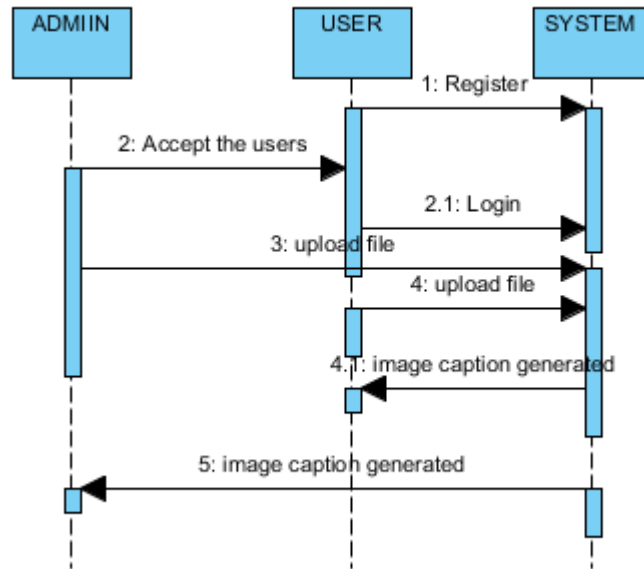
In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.



6.3 SEQUENCE DIAGRAM:

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message

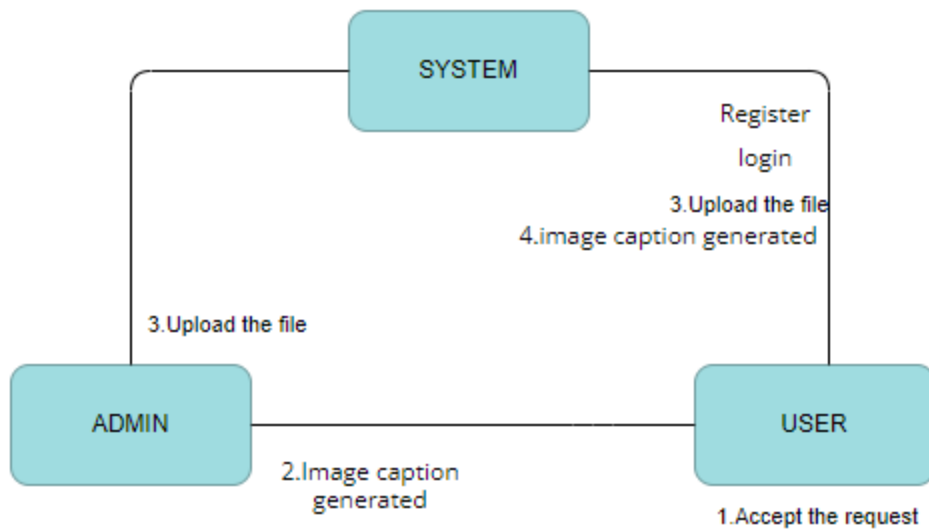
Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing



diagrams.

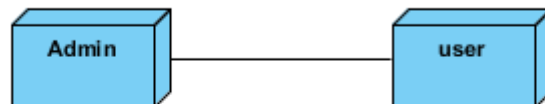
6.4 COLLABORATION DIAGRAM:

In collaboration diagram the method call sequence is indicated by some numbering technique as shown below. The number indicates how the methods are called one after another. We have taken the same order management system to describe the collaboration diagram. The method calls are similar to that of a sequence diagram. But the difference is that the sequence diagram does not describe the object organization whereas the collaboration diagram shows the object organization.



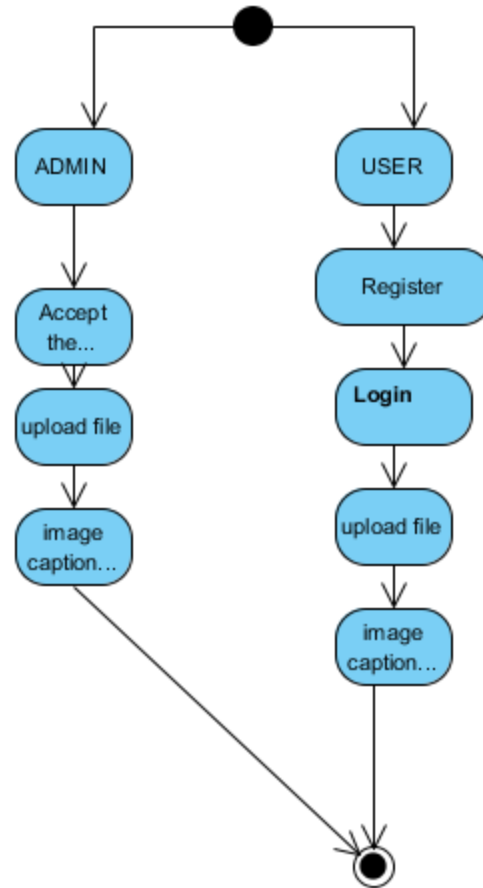
6.5 DEPLOYMENT DIAGRAM

Deployment diagram represents the deployment view of a system. It is related to the component diagram. Because the components are deployed using the deployment diagrams. A deployment diagram consists of nodes. Nodes are nothing but physical hardware's used to deploy the application.



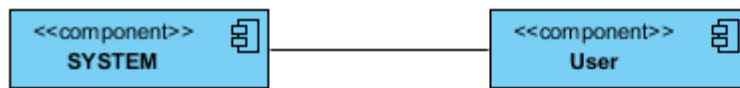
6.2.6 ACTIVITY DIAGRAM:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.



6.2.7 COMPONENT DIAGRAM.

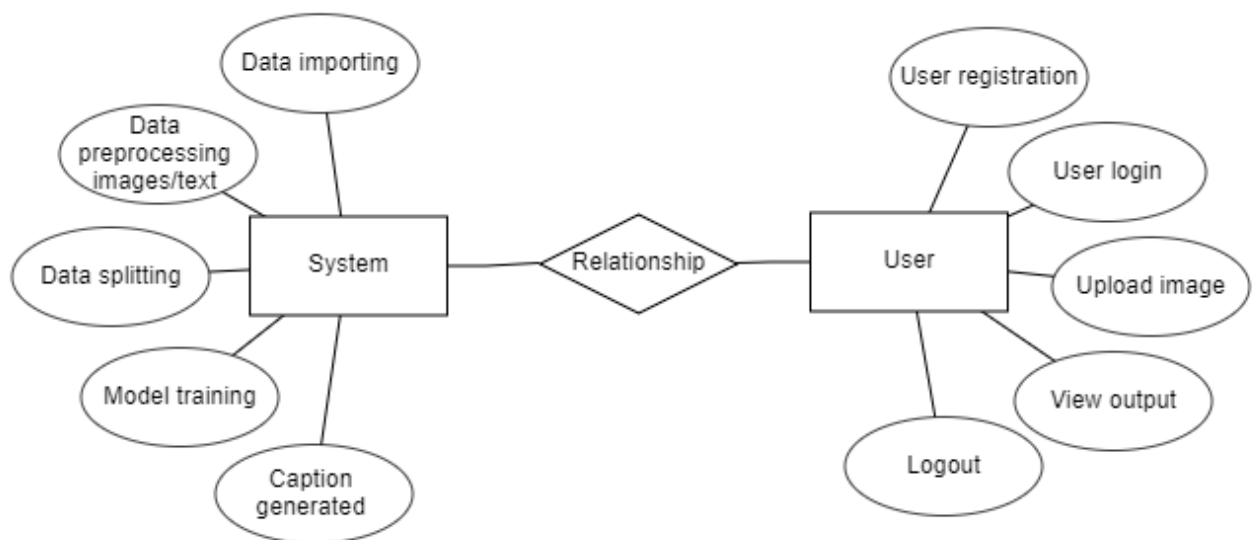
A component diagram, also known as a UML component diagram, describes the organization and wiring of the physical components in a system. Component diagrams are often drawn to help model implementation details and double-check that every aspect of the system's required functions is covered by planned development.



6.2.8 ER DIAGRAM:

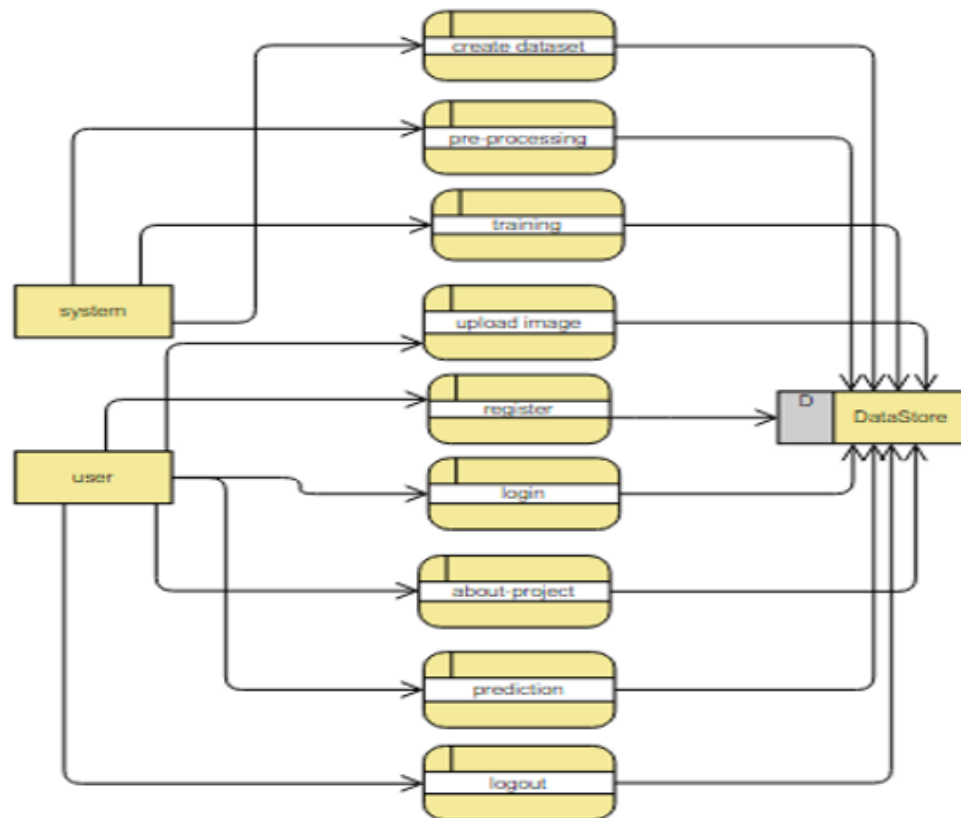
An Entity–relationship model (ER model) describes the structure of a database with the help of a diagram, which is known as Entity Relationship Diagram (ER Diagram). An ER model is a design or blueprint of a database that can later be implemented as a database. The main components of E-R model are: entity set and relationship set.

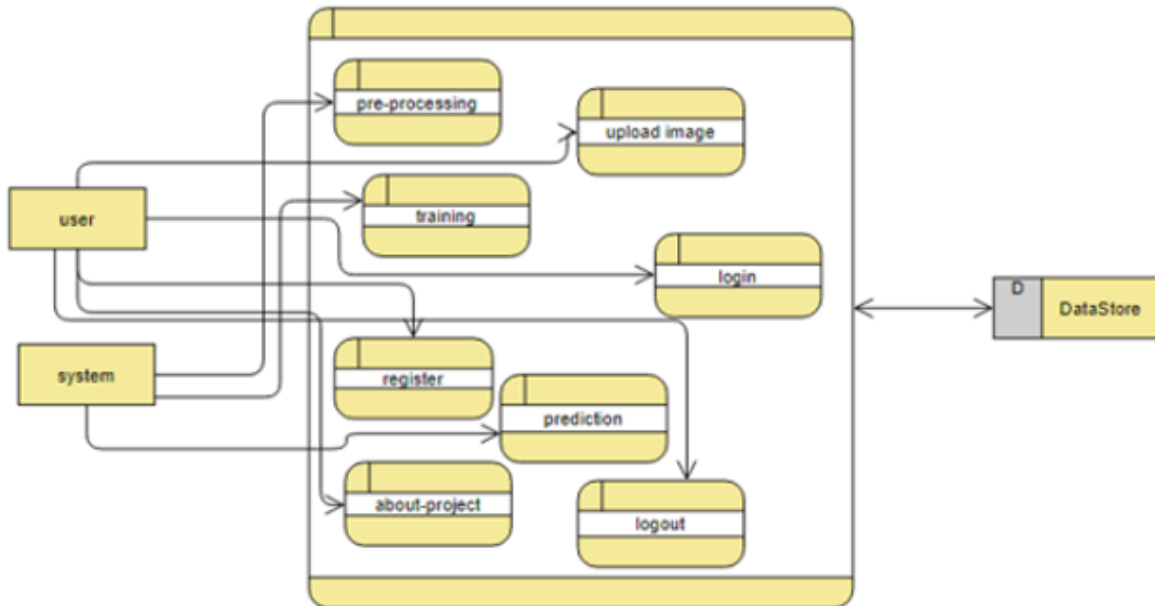
An ER diagram shows the relationship among entity sets. An entity set is a group of similar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database. Let’s have a look at a simple ER diagram to understand this concept.



6.3 DFD DIAGRAM:

A Data Flow Diagram (DFD) is a traditional way to visualize the information flows within a system. A neat and clear DFD can depict a good amount of the system requirements graphically. It can be manual, automated, or a combination of both. It shows how information enters and leaves the system, what changes the information and where information is stored. The purpose of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communications tool between a systems analyst and any person who plays a part in the system that acts as the starting point for redesigning a system.





7. IMPLEMENTATION AND RESULTS

7.1 MODULES

System:

1.1 Create Dataset:

The dataset containing images and text data of the desired objects to be captioned is split into training and testing dataset with the test size of 20-30%.

1.2 Pre-processing:

Resizing and reshaping the images into appropriate format to train our model.

1.3 Training:

Use the pre-processed training dataset is used to train our model using RESNET-50 and LSTM algorithm

2.User:

2.1 Register

The user needs to register and the data stored in the database.

2.2 Admin login

Admin logs in into the administrator login and views the user registered list, once he accepts the user data only then the user will be allowed to login.

2.3 Login

A registered user can login using the valid credentials to the website to use an application.

2.4 About-Project

In this application, we have successfully created an application which takes to classify the images.

2.5 Upload Image

The user has to upload an image which needs to be captioned the images.

2.6 Prediction

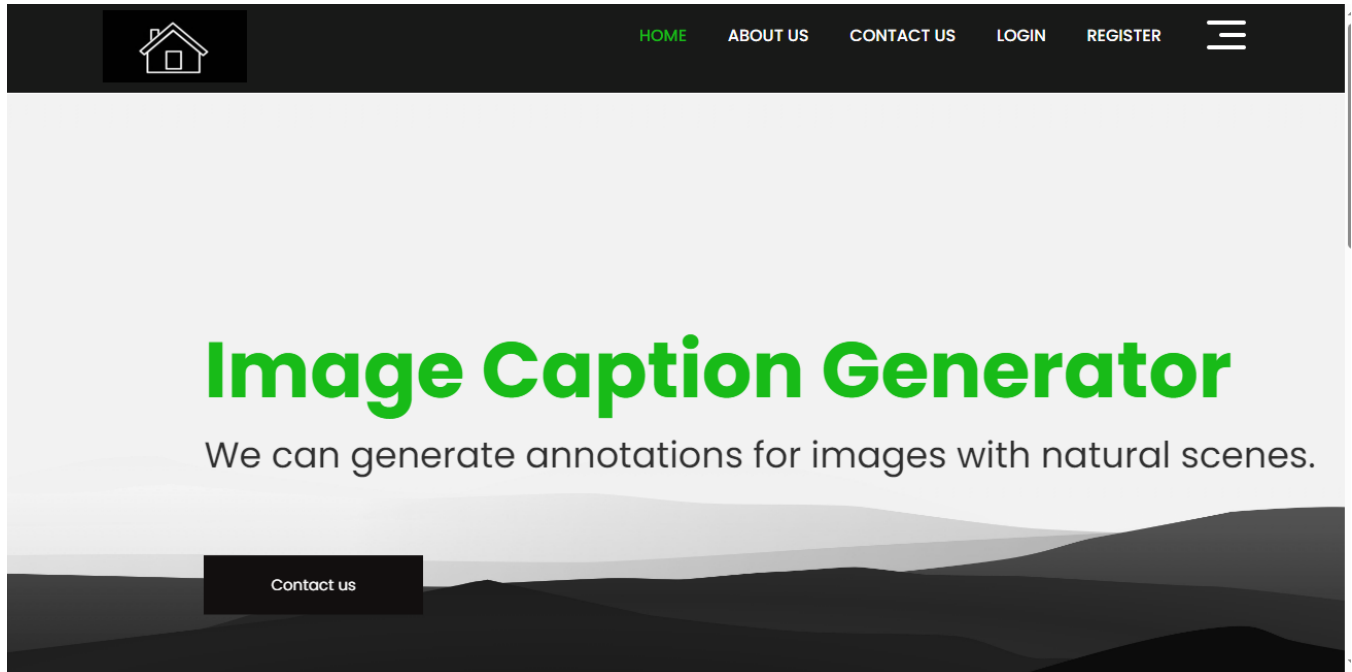
The results of our model will display the caption of the image we have assigned to it.

2.7 Logout

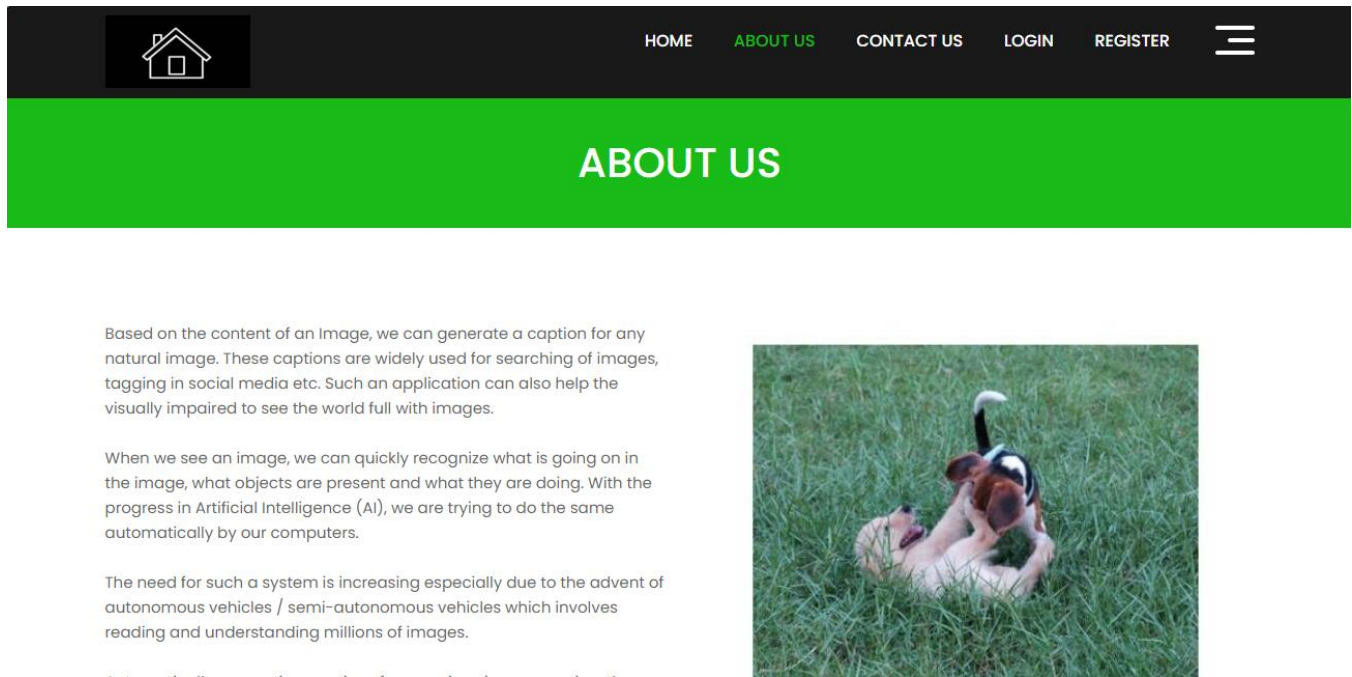
Once the prediction is over, the user can logout of the application.

RESULTS:



Home page: This is the home page where we land after clicking on the link



About Page: here we have a slight description about the project



Register page: Here User registers himself

HOMEABOUT USCONTACT USLOGINREGISTER

JOIN TODAY

Username



Email

Password

Confirm Password

Sign Up

Login Page: here user logs in with the credentials they registered with

HOMEABOUT USCONTACT US**LOGIN**REGISTER

LOG IN

Email

admin@test.com

Password

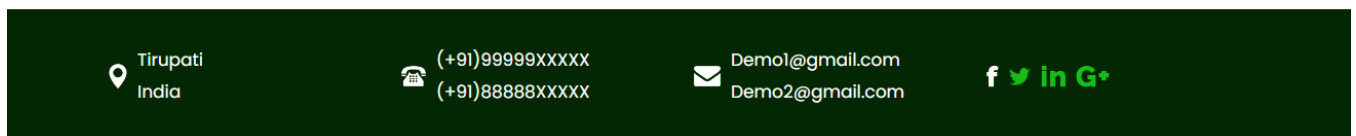
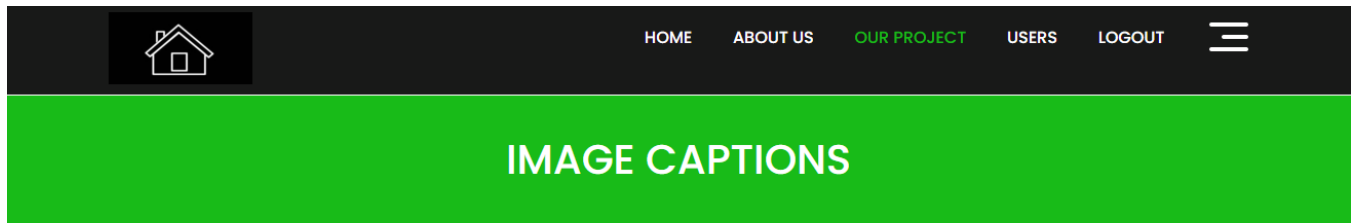
.....

☐ Remember Me

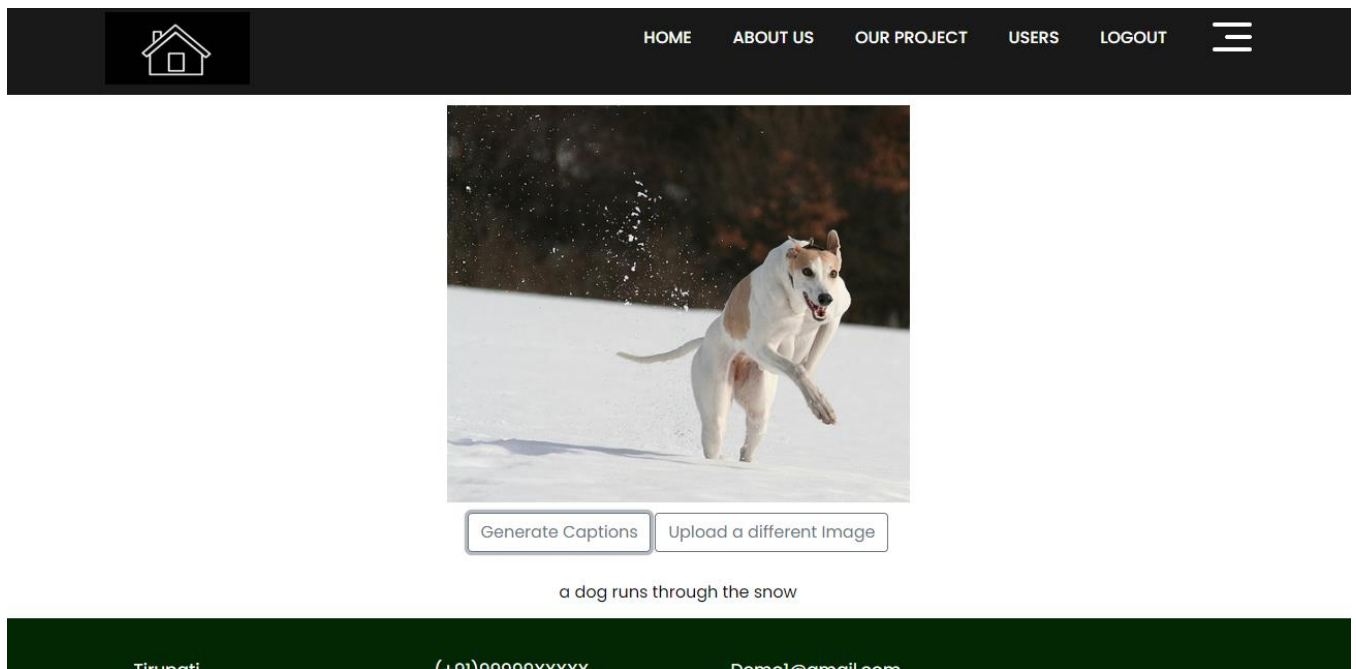
Log In

Forgot Password?

Upload page: Here user uploads the image and caption is generated



Result: Here we get the results classified



10. CONCLUSION:

The image caption generator employing a combination of Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks has proven to be a powerful and effective solution for the task of generating descriptive captions for images. The CNN-LSTM model demonstrated its ability to extract relevant features from images through the CNN layers, capturing spatial information, and then effectively utilized LSTM layers to sequence and generate coherent and contextually relevant captions. The integration of these two architectures addresses the challenges of image understanding and natural language generation, showcasing the synergy between visual perception and sequential data processing. This project not only highlights the potential of deep learning in multimodal tasks but also underscores the significance of combining specialized neural networks to achieve superior performance in complex tasks such as image captioning.

11. FUTURE WORK

Several avenues for future work can enhance and expand upon the image caption generator using CNN and LSTM. Firstly, exploring advanced architectures, such as attention mechanisms, transformer models, or pre-trained language models like BERT, could further improve the model's ability to capture intricate relationships between visual and textual information. Additionally, incorporating a larger and more diverse dataset for training can enhance the model's generalization and enable it to describe a broader range of images accurately. Fine-tuning the model for specific domains or tasks could also be valuable, allowing the generator to specialize in areas like medical imaging or satellite imagery. Furthermore, investigating techniques to make the model more interpretable and controllable could contribute to better understanding and steering the captioning process. Lastly, deploying the model in real-world applications and gathering user feedback would provide insights into its practical usability and potential areas for refinement.

12. REFERENCES

- [1] Show and Tell: A Neural Image Caption Generator by Oriol Vinyals, Alexander Toshev, Samy Bengio, Dumitru Erhan et al. CVPR 2015
- [2] Neural Image Caption Generation with Visual Attention by Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan ICML 2015

- [3] Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering by Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen (2019)
- [4] Image Captioning with Semantic Attention by Qi Wu, Chunhua Shen, Anton van den Hengel. (CVPR 2017)
- [5] DenseCap: Fully Convolutional Localization Networks for Dense Captioning by Vdovichenko et al. Justin Johnson, Andrej Karpathy, Li Fei-Fei, CVPR 2016
- [6] Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2015). Show and Tell: A Neural Image Caption Generator. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- [7] Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhutdinov, R., & Zemel, R. (2015). Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. In International Conference on Machine Learning (ICML).
- [8] Mao, J., Xu, W., Yang, Y., Wang, J., & Huang, Z. (2014). Deep Captioning with Multimodal Recurrent Neural Networks (m-RNN). In International Conference on Learning Representations (ICLR).
- [9] Donahue, J., Hendricks, L. A., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., & Darrell, T. (2015). Long-term recurrent convolutional networks for visual recognition and description. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- [10] Karpathy, A., & Fei-Fei, L. (2015). Deep Visual-Semantic Alignments for Generating Image Descriptions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- [11] Chen, X., & Lawrence, Zitnick, C. L. (2015). Mind's Eye: A Recurrent Visual Representation for Image Caption Generation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- [12] Chen, X., & Lawrence Zitnick, C. (2017). Learning to See by Moving. In Proceedings of the IEEE International Conference on Computer Vision (ICCV).

- [13] Johnson, J., Karpathy, A., & Fei-Fei, L. (2016). DenseCap: Fully Convolutional Localization Networks for Dense Captioning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- [14] Wu, Q., Shen, C., & Dick, A. (2016). Image Captioning and Visual Question Answering Based on Attributes and External Knowledge. IEEE Transactions on Pattern Analysis and Machine Intelligence.
- [15] Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In Advances in Neural Information Processing Systems (NeurIPS).
- [16] Xu, J., Mei, T., Yao, T., & Rui, Y. (2015). MSR-VTT: A Large Video Description Dataset for Bridging Video and Language. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- [17] Fang, H., Gupta, S., Iandola, F., Srivastava, R. K., Deng, L., Dollár, P., ... & He, K. (2015). From captions to visual concepts and back. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- [18] Chen, L., Zhang, H., Xiao, J., Nie, L., Shao, J., Liu, W., & Chua, T. S. (2017). SCA-CNN: Spatial and Channel-wise Attention in Convolutional Networks for Image Captioning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- [19] Wang, J., Yang, Y., Mao, J., Huang, Z., & Yuille, A. L. (2016). Cnn-rnn: A unified framework for multi-label image classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- [20] Yang, Z., He, X., Gao, J., Deng, L., & Smola, A. (2016). Stacked attention networks for image question answering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).