BMS COLLEGE OF ENGINEERING

(Autonomous Institute, Affiliated to VTU, Belagavi)

DEPARTMENT OF MACHINE LEARNING

(UG Program: B.E. in Artificial Intelligence and Machine Learning)

Image captioning using CNN,LSTM,RNN

Presented By,
<LOHITHA.T & 1BM21AI060>
<MEDHA HEGDE & 1BM21AI066>
<CHAITHRA A & 1BM22AI400>
<RAJESHWARI DM & 1BM22AI407>

Semester & Section: **4C**

In-Charge:
Prof Kusha K R
Assistant Professor
Department of Machine Learning
BMS College of Engineering

Agenda

- Introduction
- Open Issues
- Problem Statement
- Proposed Architecture
- Functional & Non-Functional Requirements
- Methodology
- Progress in Project so far
- Testing and Validation
- Conclusion
- References
- Acknowledgement (if necessary)

Introduction:

Image Captioning: Bridging Language and Vision

- An AI technology that generates descriptive textual captions for images.
- Enabling computers to understand and communicate the content of visual data.
- Combines advances in computer vision and natural language processing (NLP).

Captioning. A need?

Yes because :

- it enhances accessibility and comprehension by providing a textual description of visual content
- making it inclusive for those with visual impairments and improving user engagement.
- It also has practical applications in automating image organization and retrieval, aiding in content moderation
- assisting in the navigation of vast image datasets.

Open Issues Addressed:

- Training Data Efficiency
- Ambiguity Handling
- Error Handling
- Monitoring and Maintenance
- Data Privacy and Security
- User Customization

Problem statement

Generate caption for

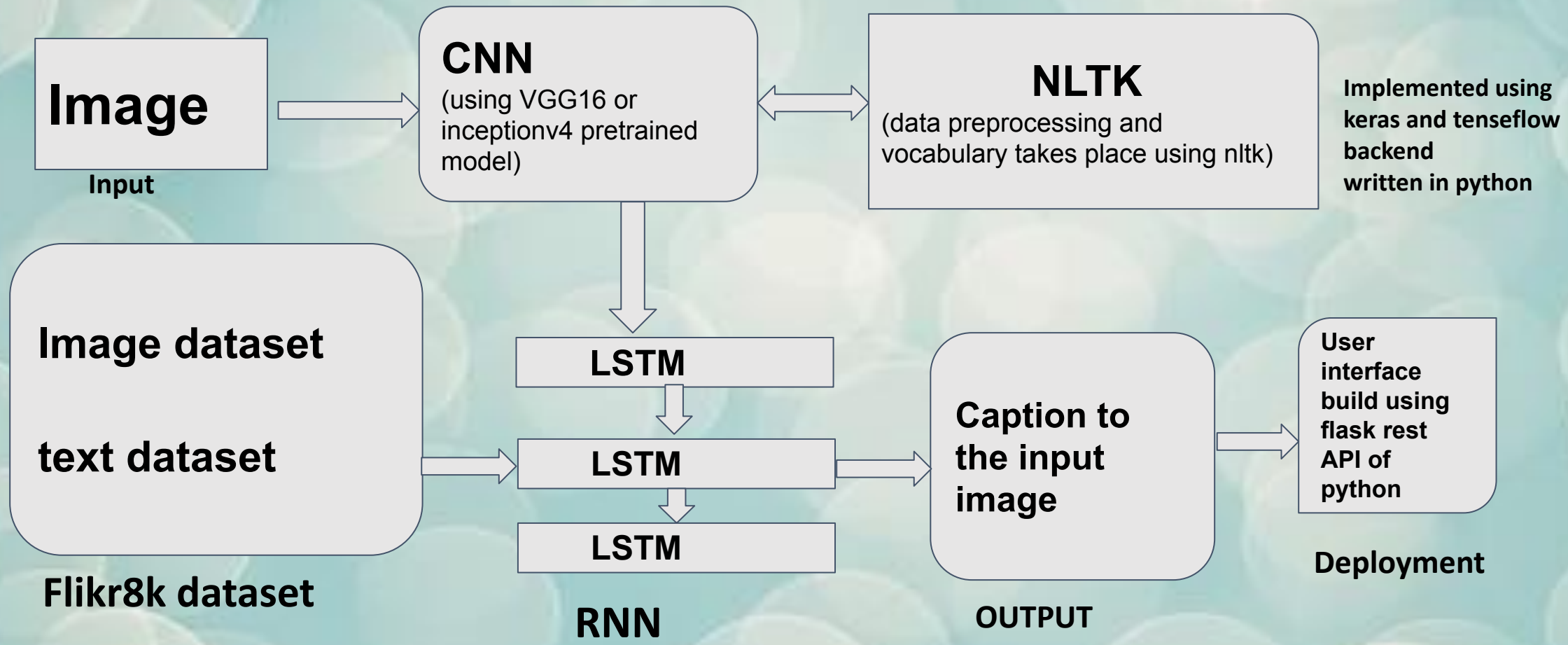
a given image

Solution - Model Development:

Building a deep learning model that combines a pre-trained CNN (e.g., ResNet) for image feature extraction with an LSTM/RNN for generating captions.

Train the model on a dataset of image-caption pairs, optimizing it to produce accurate and coherent captions

Proposed Architecture



Functional requirements

.

Image processing

Model training

User interface

Caption decoding

Flask API

scalability

Non functional requirements

.

Accuracy

Robustness

Security

Usability

Model updates

Monitoring and logging

Other Requirements

- Hardware Requirements:

RAM: 4 GB (minimum)
Hard disk: 500 GB

- Software Requirements:

Operating System : Windows (above 7 64-bit), Linux and MAC
Web Interface : Flask Rest API (Python Web Framework)
Programming Language: Python
Libraries : Tensorflow, Keras, Numpy, PIL, Flask-python, captionBot
Browser: Chrome , Firefox

What this model does basically??

Basically the model or the machine is supposed to be made such as it has to predict what is happening in the particular image.

INPUT	OUTPUT
AN IMAGE	CAPTION FOR THAT IMAGE

How does it do that??

Using a pre trained Model for object detection

|

Object Detection

|

Sentence Generation

|

Rank Based Caption Retrieval

|

Display Output

AND THE METHODOLOGY BEGINS:

Flask Web Framework: Flask is used to create a web-based interface for users to upload images and receive captions.

User Interaction: Users upload images through the Flask interface, triggering specific routes for image processing.

Image Processing: Uploaded images are processed through the pre-trained CNN to extract visual features.

Transfer Learning: We reuse a pre-trained model for object detection to enhance predictions on a new task through transfer learning.

Caption Generation: Extracted features are then used by the LSTM captioning model to generate captions.

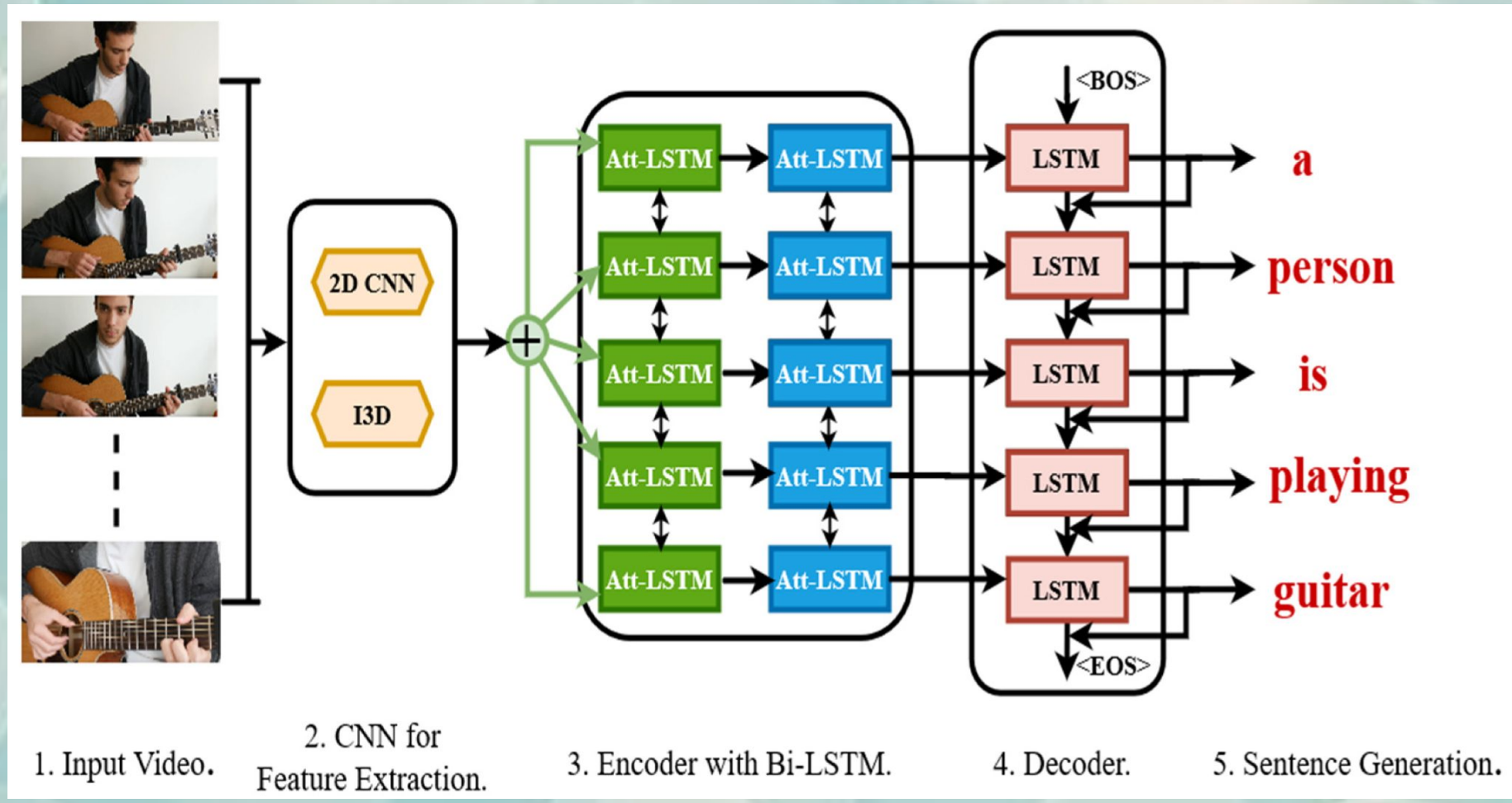
Data Preprocessing: Preprocessing involves resizing, normalizing images, and tokenizing captions.

Training: The model learns to map visual features to captions, leveraging LSTM's ability to capture sequential dependencies

Inference: Trained model generates captions for new images by passing features from CNN to LSTM, considering context for each word.

Evaluation: Captions are assessed using metrics to measure their similarity to human-written captions

- Flask takes the generated caption and **the top ranked one** and displays it on the web interface, making it accessible to the user.



Current status and progress of model:

The web interface for image captioning using deep learning is now complete, with both the front-end and back-end components fully integrated via Flask.

However, some **modifications are required in the front-end to enhance its functionality and UI** further and it's important to note that **further model training is needed to improve accuracy**, as the current accuracy stands at 75%.

To conclude-

In summary, CNN-LSTM image captioning holds great promise, with ongoing improvements and diverse applications on the horizon.

- Effective Fusion
- Practical Utility
- Challenges and Progress
- Human-AI Collaboration
- Multilingual Potential:

Automated Image Captioning

Choose File lilgirl.jpeg

Predict Caption

Predicted caption

a girl wearing orange and white shirt is playing on a red toy . .

Google Chrome

Predicted Caption

a group of people are talking to a crowd of guys in an photograph with black urban hands in front

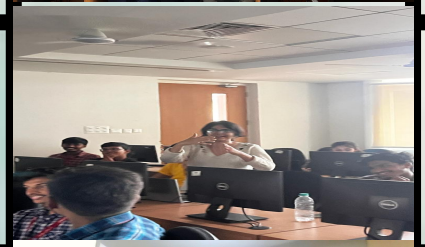
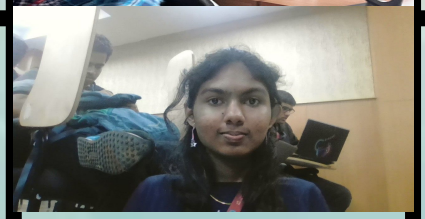
App.py running in localhost 5000

```
 * Serving Flask app 'app'
 * Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment.
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
 * Restarting with stat
++++++++++++++++++++++++++++++++++++++++++++++++++
vocabulary loaded
================================================================================
======================
MODEL LOADED
================================================================================
======================
RESNET MODEL LOADED
 * Debugger is active!
 * Debugger PIN: 668-601-333
127.0.0.1 - - [11/Sep/2023 01:16:25] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [11/Sep/2023 01:16:25] "GET /static/styles.css HTTP/1.1" 404 -
127.0.0.1 - - [11/Sep/2023 01:16:25] "GET /favicon.ico HTTP/1.1" 404 -
================================================
IMAGE SAVED
1/1 [==============================] - 1s 859ms/step
================================================
Predict Features
```

Training the model to the accuracy: (73.28%) got

```
46
Epoch 46/50
188/188 [==============================] - 13s 71ms/step - loss: 1.1727 - accuracy: 0.70
18
Epoch 47/50
188/188 [==============================] - 14s 73ms/step - loss: 1.1377 - accuracy: 0.71
08
Epoch 48/50
188/188 [==============================] - 14s 73ms/step - loss: 1.1055 - accuracy: 0.71
93
Epoch 49/50
188/188 [==============================] - 14s 73ms/step - loss: 1.0717 - accuracy: 0.72
66
Epoch 50/50
188/188 [==============================] - 14s 75ms/step - loss: 1.0449 - accuracy: 0.73
28
```

Testing and Validation

TEST CASE	GIVEN INPUT	EXPECTED OUTPUT	OBTAINED OUTPUT
1		Group of people in a room together	Group of people talking to crowd of guy
2		A girl in a lab	A woman in fancy room
3		A girl near table	A girl with baby on the table
4		A girl in front of rainbow	A girl playing wth blue color wall

Thank you !