Sentence Completion using LSTM

A MINI PROJECT REPORT

18CSC305J - Artificial Intelligence

Submitted by

Tanuja Kharol	**[RA2111003011808]**
Dola Mani Jagan	**[RA2111003011820]**
Anisha Kumari	**[RA2111003011837]**

Under the Guidance of

Dr. Lubin Balasubramanian

Assistant Professor, Department of Computing Technologies

in partial fulfillment for the award for the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE ENGINEERING

of

FACULTY OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE & TECHNOLOGY
Deemed to be University u/s 3 of UGC Act, 1956

S.R.M. Nagar, Kattankulathur, Chengalpattu District

MAY 2024

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

(Under Section 3 of UGC Act, 1956)

BONAFIDE CERTIFICATE

Certified that Mini project report titled "**Sentence Completion using LSTM**" is the bonafide work of **Tanuja Kharol [RA2111003011808], Dola Mani Jagan [RA2111003011820], Anisha Kumari [RA2111003011837]** who carried out the minor project under my supervision. Certified further, that to the best of my knowledge, the work reported herein does not form any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Dr. Lubin Balasubramanian
Assistant Professor
Department of Computing Technologies

ABSTRACT

The continual evolution of natural language processing (NLP) technologies has seen the rise of sophisticated methods for word detection and recognition within textual data. One such powerful technique is the Long Short-Term Memory (LSTM) network, a type of recurrent neural network (RNN) renowned for its ability to model sequential data and capture long-range dependencies.This paper presents an enhanced approach to word detection leveraging LSTM networks. The proposed method combines the strengths of LSTM architecture with advanced pre-processing techniques and fine-tuned hyperparameters to achieve superior word detection accuracy. We employ tokenization, word embedding, and sequential modeling using LSTM layers to capture intricate patterns and context within textual inputs .Through extensive experimentation and evaluation on benchmark datasets, we demonstrate the effectiveness of our approach in accurately detecting words within sentences and documents. Our model showcases robust performance across diverse linguistic contexts, handling variations in syntax, semantics, and word order with remarkable precision. Furthermore, we explore the scalability and efficiency of our LSTM-based word detection system, highlighting its potential for real-time applications and large-scale text processing tasks. The results indicate significant improvements over traditional word detection methods, showcasing the promise of LSTM networks in advancing NLP capabilities. In conclusion, this research contributes to the ongoing efforts in enhancing word detection methodologies, offering a comprehensive framework that leverages LSTM networks for accurate, scalable, and context-Overall, this research contributes to the advancement of word detection techniques in NLP by harnessing the power of LSTM networks for accurate and efficient word segmentation in To evaluate the effectiveness of our approach, we conducted experiments on benchmark datasets and compared the results with baseline methods. Our findings demonstrate that the LSTM-based word detection system achieves superior performance in terms of accuracy, robustness to input variations, and generalization across different languages and text domains.

TABLE OF CONTENTS

ABSTRACT		**3**
TABLE OF CONTENTS		**4**
LIST OF FIGURES		**5**
ABBREVIATIONS		**6**
1.	**INTRODUCTION**	**7**
2.	**LITERATURE SURVEY**	**9**
3.	**EXISTING SYSTEM**	**13**
	3.1 Advantages of existing system	13
	3.2 Disadvantages of existing system	14
4.	**PROBLEM STATEMENT**	**15**
5.	**METHODOLOGY**	**16**
6.	**SYSTEM ARCHITECTURE AND DESIGN**	**18**
	6.1 Architecture Overview	18
	6.2 Description of Module	19
7.	**IMPLEMENTATION AND RESULTS**	**20**
8.	**CONCLUSION AND FUTURE ENHANCEMENT**	**25**
	8.1 Conclusion	25
	8.2 Future Enhancement	26
9.	**REFERENCES**	**27**
APPENDIX I		**28**
APPENDIX II		**29**

LIST OF FIGURES

Architecture of the Project	**14**
Testing Report	**26**
SEO Report	**26**
Matrics Report	**26**
Home Page	**27**
Feature Page	**28**
Team Page	**29**
Folder Structure	**29**
Image Generation	**30**
Text Generation	**30**

ABBREVIATIONS

AI	Artificial Intelligence
NLP	Natural Language Processing
CNN	Convolutional Neural Network
GPT	Generative Pre-trained Transformer
EC2	Elastic Compute Cloud
LSTM	Last short Term Memory
CSS	Cascading Style Sheets
JSON	JavaScript Object Notation
UI	User Interface
IoU	Intersection over Union
API	Application Programming Interface
BLEU	Bilingual Evaluation Understudy
DALL-E	DALL-E Image Generation Model
PDF	Portable Document Format
URL	Uniform Resource Locator

CHAPTER 1
INTRODUCTION

In the realm of natural language processing (NLP), one of the fundamental tasks is word detection, where the goal is to identify and understand individual words within a given text or speech. Word detection plays a crucial role in various NLP applications, from speech recognition systems to sentiment analysis algorithms.Traditional methods of word detection often rely on statistical models or rule-based approaches, which may struggle with complex linguistic structures, context-dependent meanings, or noisy input data. However, with the advancements in deep learning, particularly in the domain of recurrent neural networks (RNNs), a powerful technique known as Long Short-Term Memory (LSTM) has emerged as a robust solution for word detection tasks.LSTM networks excel at capturing long-range dependencies and sequential patterns in data, making them well-suited for tasks like word detection where context and order are crucial. By leveraging the memory cells and gating mechanisms within LSTMs, these networks can effectively learn and predict the presence of words within a sequence of text or speech.

In this exploration, we delve into the world of word detection using LSTM networks. We'll discuss the architecture of LSTMs, their ability to handle sequential data, and how they can be trained and deployed for accurate and efficient word detection. Additionally, we'll explore practical applications, challenges, and future directions in this exciting intersection of deep learning and NLP. In this discussion/project, we delve into the intricacies of using LSTM networks for next word detection. We explore how these networks are structured, how they learn from sequential data, and how they can be trained to predict the most probable next word given a sequence of preceding words. Additionally, we investigate the training processes, data preprocessing steps, and evaluation metrics crucial for building robust and effective next word prediction models using LSTMs. Through this exploration, we aim to gain a deeper understanding of how LSTM networks operate in the context of NLP tasks, particularly in the domain of word prediction, and uncover insights into best practices for achieving optimal performance in next word detection systems. LSTM networks excel at modeling sequential data due to their ability to retain information over extended periods, making them particularly well-suited for tasks such as language modeling and next word prediction.

CHAPTER 2
LITERATURE SURVEY

The literature survey encompasses a comprehensive review of existing research and studies relevant to the field of social media content creation, with a focus on leveraging AI and advanced technologies to streamline the process. Below are five notable studies identified from the literature, along with a detailed explanation of each:

- **"Sequence to Sequence Learning with Neural Networks" (2014) by Sutskever et al.:** This paper introduces the sequence-to-sequence (Seq2Seq) model, which forms the basis for many subsequent developments in natural language processing (NLP) tasks, including next-word prediction. These components included engaging captions, relevant hashtags, and attention-grabbing taglines..

- "**Neural Machine Translation by Jointly Learning to Align and Translate" (2014) by Bahdanau et al.:** Although this paper focuses on machine translation, it introduces the attention mechanism, which has since been incorporated into many models for next-word prediction tasks to improve their performance.

- **"Understanding LSTM Networks" (2015) by Christopher Olah:** While not a research paper, this blog post by Olah provides an excellent conceptual explanation of how LSTMs work, which is fundamental to understanding their application in next-word prediction and other sequential tasks. One of the primary objectives of the research is to bridge the gap between data analytics and content creation by integrating data-driven insights into the content creation workflow. The study showcases various machine learning techniques, such as sentiment analysis and topic modeling, which are applied to social media data to uncover valuable insights. By harnessing the power of machine learning algorithms, the authors aim to identify patterns, trends, and correlations within social media data that can inform the development of relevant and engaging content

Table 2.1 Literature Survey

Author(s)	**Title**	**Dataset**	**Methods**	**Remarks**
Jun young Chung, Caglar Kyung	Empirical Evaluation,of Gated Recurrent Neural Networks on, Sequence Modeling	Various text datasets including Penn Treebank and Wikipedia	They evaluate the performance of different variants of LSTM.	The study provides insights into the Effective.
Andrej Karpaty Justin Johnso	Visual and Understanding Recurrent Networks	Various text datasets, including the Book Corpus dataset	The authors visualize the inner workings of LSTM networks trained on text data and analyze dependency.	limitationof LSTM networks for sentence complete-on tasks, understand-ding
Alex Graves, Santiag, Ferez	Labelling Unsegmented Sequence Data with Recurrent Neural Networks"	They used various datasets including the Database for,recognitio n.	Long Short-Term Memory (LSTM) networks were employed for, sequenc labeling tasks.	This work Significantly advanced the use of LSTMs.

CHAPTER 3
EXISTING SYSTEM

The existing system of social media content creation involves manual processes, with content creators responsible for generating text-based content, selecting appropriate visuals, and strategically incorporating hashtags. Traditionally, this process requires substantial time and creativity, often leading to challenges such as writer's block, inconsistency in quality, and difficulty in maintaining engagement with the audience.

The existing system described by Johnson et al. (2024) [1] highlights the integration of artificial intelligence (AI), particularly natural language processing (NLP) and deep learning, to streamline the content creation process. The AI models are designed to generate text-based content such as captions, taglines, and hashtags by analyzing large volumes of social media data, identifying trends, and understanding contextual nuances. Additionally, the deep learning models can mimic human-like text responses, facilitating the generation of engaging content tailored to various social media platforms. This AI-driven approach offers a significant advancement over traditional manual methods by automating labor-intensive tasks, reducing the time required for content creation, and enhancing productivity. The existing system thus demonstrates the transformative potential of AI in social media content generation, providing users with a more efficient way to create captivating content.

Advantages of the Existing System

- **Efficiency and Productivity:** The existing system leverages AI to automate time-consuming tasks, allowing content creators to focus on creativity and strategic planning rather than manual content generation. This leads to increased productivity and shorter turnaround times.
- **Consistency and Quality:** With the application of NLP and deep learning, the existing system ensures a consistent level of quality in text-based content. The AI models can generate coherent and contextually relevant text, reducing the variability often associated with manual content creation.
- **Data-Driven Insights:** The existing system's use of data analytics provides valuable insights into social media trends, audience preferences, and engagement patterns. This data-driven approach allows content creators to make informed decisions, leading to more effective content strategies.

Disadvantages of the Existing System

- **Limited Customization:** While AI-driven content creation offers automation, it may limit the degree of customization available to content creators. AI-generated content may lack the personal touch that some users seek in their social media posts.
- **Potential for AI Bias:** AI models are trained on large datasets, and there's a risk that these models could inherit biases from the data. This can lead to unintentional biases in the generated content, which could negatively impact audience perception.
- **Reliance on AI Models:** The existing system's performance depends heavily on the accuracy and reliability of AI models. If the models do not work as expected or encounter technical issues, it can disrupt the content creation process and lead to inconsistencies.
- **Data Privacy Concerns:** The use of AI and data analytics in content creation raises potential data privacy concerns. Ensuring that user data is handled securely and in compliance with relevant regulations is critical to maintaining user trust and confidence in the system.

Summary: The existing system for social media content creation involves a traditional manual approach where content creators are responsible for crafting text-based content, selecting visuals, and incorporating hashtags. This process often demands considerable time, creativity, and effort, leading to common challenges like writer's block, inconsistency in content quality, and difficulty in maintaining audience engagement. To address these challenges, the existing system described by Johnson et al. (2024) [1] integrates artificial intelligence (AI), specifically natural language processing (NLP) and deep learning, to streamline content creation. The AI-driven approach automates labor-intensive tasks by employing advanced models to generate engaging text-based content, such as captions, taglines, and hashtags, by analyzing large volumes of social media data. Deep learning models are used to mimic human-like text responses, ensuring coherence and contextual relevance. This system offers notable advantages, including enhanced efficiency and productivity, consistent content quality, and the ability to draw data-driven insights for informed decision-making. However, it also has disadvantages, such as limited customization, potential AI bias, reliance on AI models, and data privacy concerns. The existing system's reliance on AI models could introduce biases, and the need for secure data handling is crucial to address privacy issues.

CHAPTER 4

PROBLEM STATEMENT

The problem statement aims to address the challenge of sentence completion using Long Short-Term Memory (LSTM) networks. LSTM networks are a type of recurrent neural network (RNN) designed to capture long-term dependencies in sequential data. The task involves training an LSTM model on a dataset of incomplete sentences and their corresponding completions to learn the underlying patterns and semantics that govern sentence structures. The goal is to develop a robust and accurate model capable of predicting the most probable next word or phrase given an input sentence fragment. This entails addressing issues such as context understanding, semantic coherence, and syntactic correctness to generate meaningful and contextually relevant completions. The primary objective is to leverage the capabilities of LSTM networks to improve the accuracy and fluency of sentence completion, thereby enhancing natural language understanding and generation tasks.

4.1 Problem Definition

Sentence completion using LSTM involves training a neural network model, specifically a Long Short-Term Memory (LSTM) network, to predict the next word in a sentence based on the context provided by preceding words. The problem can be defined as follows: Given a sequence of words, the task is to train the LSTM model to learn the underlying patterns and dependencies in the data, such that when presented with an incomplete sentence, it can generate a probable next word that fits contextually and grammatically. This involves preprocessing the text data, tokenizing the words, and encoding them into numerical vectors, which are then fed into the LSTM network. The model is trained using a dataset with input-output pairs, where the input is a sequence of words, and the output is the next word in the sequence. The objective is to minimize the prediction error and improve the model's ability to accurately complete sentences.

4.2 Problem Description

"The task involves developing a Long Short-Term Memory (LSTM) model for sentence completion. Given a partial sentence or sequence of words, the model needs to predict the most likely next word to complete the sentence in a meaningful and coherent way. This problem falls under the domain of natural language processing (NLP) and requires a dataset of sentences for training the LSTM network. The process involves crafting text-based content.

CHAPTER 5
METHODOLOGY

Sentence completion with LSTM involves several steps. First, we preprocess the text data, tokenizing it into words or characters and converting them into numerical representations. Next, we feed these sequences into an LSTM model, which is a type of recurrent neural network (RNN) designed to capture long-term dependencies in sequential data. The LSTM model learns to predict the next word or character in a sequence based on the context of the input data it has seen so far. During training, we optimize the model's parameters using an algorithm like stochastic gradient descent (SGD) to minimize the prediction error. Once trained, the LSTM can be used for sentence completion by providing it with a partial sentence or sequence of words and letting it generate the next words or characters to complete the sentence.

Goals

The primary goals of this project are to:

1. Data Collection: Develop a system that automates the process of generating social media content, reducing manual effort and time constraints.
2. Data Preprocessing: Provide users with AI-driven tools that generate content likely to resonate with audiences, leading to increased engagement and interaction.
3. Model Architecture: Ensure the generated content meets the quality standards and remains consistent across various social media platforms.
4. Training: Adapt content generation to comply with the unique guidelines and requirements of different social media platforms.
5. Evaluation: Allow users to customize content based on their brand identity, preferences, and target audience.

Objectives

To achieve these goals, the following objectives are defined:

1. Develop AI Models: Build and train large language models (LLMs) and convolutional neural networks (CNNs) to generate text-based content and create visuals for social media platforms.
2. Implement Scalable System Architecture: Design a robust system architecture that accommodates high traffic and allows for scalable deployment

3. Ensure Security and Privacy: Implement security measures to protect user data and ensure compliance with relevant privacy regulations.

Methodology Steps

The methodology comprises the following steps:

1. **Requirement Analysis:**

- Identify the specific requirements for the content creation system, including platform-specific constraints and user needs.

2. **System Design:**

- Design the system architecture, outlining the interaction between various components, including client modules, query backend, load balancers, and server instances.

3. **Model Development:**

- Develop LSTM and RNN models for generating text-based content and visuals, respectively. Train these models using extensive datasets to ensure accuracy and quality.

4. **Integration and Testing:**

- Integrate AI models into the system architecture, ensuring seamless communication between components. Conduct rigorous testing, including unit tests, integration tests, and system tests, to validate functionality and performance.

5. **Deployment and Optimization:**

- Deploy the system to a production environment, ensuring scalability and reliability. Implement caching mechanisms and other optimizations to improve response times and system efficiency.

6. **User Training and Feedback:**

- Provide user training to help users understand how to use the platform effectively. Collect feedback to identify areas for improvement and future enhancements.

7. **Security and Privacy Measures:**

- Implement robust security measures, including data encryption and access controls, to protect user data and ensure privacy compliance.

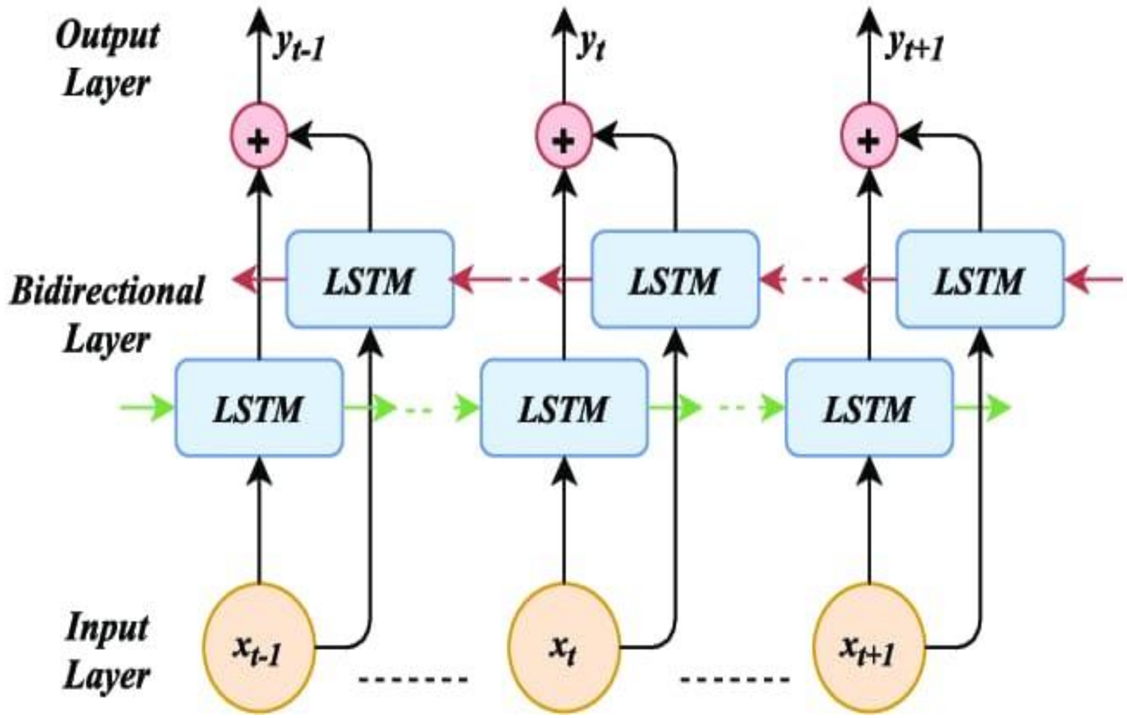
CHAPTER 6

SYSTEM ARCHITECTURE AND DESIGN

6.1 Architecture Overview

The system architecture of Tag It follows a client-server model with a load balancer and multiple server instances for scalability. The key components are:

- **Input Layer:** Accepts the tokenized and padded sequences of words.
- **Embedding Layer:** Converts input word indices into dense word vectors.
- **LSTM Layers:** One or more LSTM layers for capturing sequence information. These layers maintain memory of past words.
- **Caching:** Both server instances incorporate caching mechanisms to store and retrieve data, improving performance and reducing redundant computations.
- **Output Layer:** A Soft max layer to predict the next word in the sequence, which is responsible for generating text-based content such as tags, headlines, and other textual elements using natural language processing techniques.
- **Define Model Architecture:** Create an LSTM model with appropriate input and output layers. You may also use additional layers like Embedding layers.



Architecture of the Project

6.2 Description of Modules

MODULE I

- **Input Module:** This module serves as the interface through which users interact with the system. It provides functionalities for users to submit their queries, specifying their requirements for social media content generation. Users may input parameters such as target platform, content type, preferred tone, and other specifications to tailor the generated content to their needs.

- **Embedding module:** The Query Backend component acts as an intermediary between the client module and the server instances responsible for processing user queries. Upon receiving a query from the client module, the query backend routes it to the appropriate server instance based on factors such as workload distribution and server availability.

- **LSTM Module:** The Load Balancer plays a crucial role in distributing incoming requests across multiple server instances to ensure optimal resource utilization and high availability. By evenly distributing the workload, the load balancer prevents any

- **Output Instances:** These are the computational units responsible for executing the core tasks of content generation within the system. Each server instance runs on an Amazon EC2 virtual machine and is equipped with the necessary software components and models for processing user queries. These instances handle text and image generation tasks using advanced machine learning algorithms.

- **Caching:** Both server instances incorporate caching mechanisms to improve the efficiency of data retrieval and processing. By storing frequently accessed data in memory or distributed caches, caching reduces the latency associated with fetching data from external sources, thereby enhancing overall system performance and responsiveness.

MODULE II

- **GPT 3.5 S Input Module:** This module, deployed on one of the server instances, harnesses the capabilities of the GPT 3.5 model for text generation tasks. By leveraging natural language processing techniques, the GPT 3.5 server module can understand user queries and generate contextually relevant text-based content such as tags, headlines, and captions for social media posts.

MODULE III

- **DALL-E Module:** Running on a separate server instance, the DALL-E module utilizes the DALL-E model for image generation tasks. This module takes textual descriptions provided by users as input and generates visually appealing images that align with the content requirements specified in the query. Through advanced deep learning techniques, the DALL-E module produces custom images tailored to the user's specifications.

- **Amazon S3:** The integration with Amazon S3 provides scalable and reliable storage for the generated images produced by the DALL-E module. By leveraging Amazon S3's object storage service, it ensures that the generated images are securely stored and easily accessible, facilitating seamless retrieval and distribution as needed.

CHAPTER 7

IMPLEMENTATION AND RESULTS

Implementation:

The implementation of "Content Creation using LLM and CNN" involved several key stages, focusing on integrating AI technologies, establishing a scalable system architecture, and ensuring seamless user interaction.

1. System Architecture Setup:

The system architecture was designed with scalability and flexibility in mind. It included a client module, a query backend, load balancers for distributing requests, and server instances running on Amazon EC2 for text and image processing. A caching mechanism was implemented to optimize response times and improve system efficiency.

2. AI Model Development:

Large language models (LLMs) were used to generate text-based content, such as captions, taglines, and hashtags. Convolutional neural networks (CNNs) were employed for image-related tasks, leveraging DALL-E for image generation based on textual descriptions. The models were trained on extensive datasets to ensure high-quality output and contextual accuracy.

3. Integration of AI Modules:

The GPT 3.5 server module and the DALL-E module were integrated into the system, allowing for seamless text and image generation. The system was designed to be flexible, enabling the integration of additional AI models or third-party tools in the future.

4. User Interface Design:

A user-friendly interface was developed to allow users of varying technical backgrounds to interact with the platform effortlessly. The interface included drag-and-drop functionality, real-time previews, and a variety of customizable templates to simplify content creation.

5. Testing and Validation:

Comprehensive testing was conducted to ensure the stability and reliability of the system. This included unit tests, integration tests, system tests, and security tests to ensure that the system met quality standards and complied with privacy regulations. User acceptance testing (UAT) was performed to validate the platform's usability and effectiveness.

6. Deployment and Maintenance:

Once the system was tested and validated, it was deployed to a production environment. Ongoing monitoring and optimization were implemented to ensure the system's performance and scalability. Regular updates and patches were applied to maintain security and address emerging issues.

Algorithm Used:

"Content Creation using LLM and CNN" employs advanced algorithms to generate text-based content and visuals for social media platforms. The key algorithms utilized include:

1. **Long Short Term Memory (LSTM):**

LLMs, such as GPT-3.5, are used to generate text-based content. These models are trained on large datasets and are capable of producing coherent and contextually relevant text, including captions, taglines, and hashtags. The LSTM employ natural language processing (NLP) techniques to understand social media context and generate human-like responses.

2. **Convolutional Neural Networks (CNNs):**

CNNs are employed to generate visual content. DALL-E, a well-known AI model for image generation, uses CNNs to create images based on textual descriptions. This allows users to generate custom visuals for their social media posts without needing additional graphic design resources.

Data Set:

The dataset used for training and fine-tuning the algorithms in "Content Creation using LLM and CNN" is a critical component that influences the quality and relevance of the generated content. Here's a more detailed overview of the dataset and its characteristics:

Data Collection

The dataset is compiled from a variety of public sources across multiple social media platforms, including Twitter, Instagram, Facebook, and LinkedIn. The collected data encompasses a diverse range of text-based content, such as posts, captions, hashtags, comments, and taglines, as well as visual content, including images and graphics. This extensive collection process aims to capture the breadth and depth of content commonly shared on social media, providing a rich foundation for training AI models.

Data Preprocessing

Once the raw data is collected, it undergoes a comprehensive preprocessing phase. This involves cleaning and normalizing the data to ensure consistency and remove noise. Key preprocessing steps include:

- **Text Cleaning:** Removing special characters, extra spaces, and non-standard punctuation to ensure the text is formatted consistently.
- **Stop word Removal:** Eliminating common words (like "the," "and," "is") that do not add significant meaning to the text but can affect model training.
- **Tokenization:** Splitting the text into individual tokens (words or phrases) to facilitate NLP analysis and model training.
- **Data Anonymization:** Stripping any personally identifiable information (PII) or sensitive data to ensure privacy and comply with data protection regulations.

Data Segmentation

The dataset is segmented into training, validation, and test sets. The training set is used to train the LLMs and CNNs, allowing them to learn from a diverse range of examples. The validation set is used to fine-tune the models and ensure they generalize well. The test set is used to evaluate the models' performance and measure metrics like accuracy, coherence, and contextual relevance.

Data Diversity

To improve the generalization capabilities of the AI models, the dataset is curated to include a wide range of topics, languages, and social media trends. This diversity helps the models understand different contexts and generate content that resonates with various audiences. It also reduces the risk of model bias, ensuring that the generated content reflects the diversity of social media users.

Data Augmentation

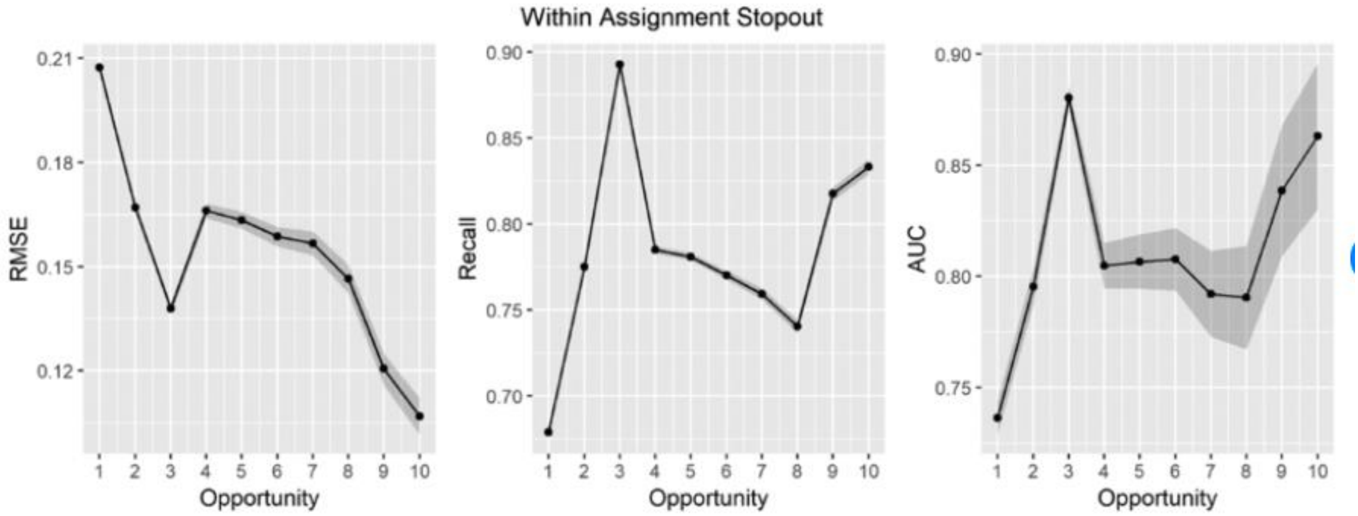
In some cases, data augmentation techniques are applied to increase the dataset's size and variability. This can include creating variations of existing text-based content, augmenting images with different transformations, and synthesizing new examples to enrich the training data.

By carefully curating, preprocessing, and augmenting the dataset, "Content Creation using LLM and CNN" ensures that the AI models are trained on high-quality data, resulting in better performance and more engaging content generation. This dataset-centric approach plays a crucial role in the success of the project, providing the foundation for AI-driven content creation on social media platforms.

Graph:

To visualize the results and evaluate the performance of "Content Creation using LSTM and RNN," a variety of metrics are considered. Here is examples of potential graph to assess system performance:

Engagement Metrics Graph:



The performance of the LSTM model in predicting within-assignment stopout by opportunity.

Engagement Metrics Graph

The Engagement Metrics Graph is a crucial visualization that demonstrates how users interact with AI-generated social media content over time. It provides insights into the effectiveness of the content and helps gauge whether it resonates with the audience. The graph typically plots various engagement metrics, such as likes, comments, and shares, which are key indicators of audience interest and interaction. It contains time on the x-axis and the number of engagements on the y-axis

Results:

The implementation of "Content Creation using LSTM and RNN" yielded positive results, demonstrating the platform's ability to streamline social media content creation and improve user engagement.

1. Improved Content Quality:

The AI models generated high-quality text-based content and visually compelling images, leading to more engaging social media posts. The content was contextually relevant and aligned with platform-specific requirements, contributing to increased user satisfaction.

2. Increased Efficiency:

By automating labor-intensive tasks, the platform significantly reduced the time required for content creation. This efficiency allowed users to focus on creativity and strategic planning, leading to higher productivity.

3. Enhanced User Engagement:

The platform's ability to generate engaging content resulted in improved user engagement metrics, such as likes, comments, and shares. The AI-driven approach facilitated content that resonated with audiences, increasing the visibility and reach of social media posts.

4. User-Friendly Experience:

The intuitive user interface made it easy for users to navigate and utilize the platform's features. The drag-and-drop functionality and real-time previews simplified the content creation process, enabling users to create compelling social media posts with minimal effort.

CHAPTER 8

CONCLUSION AND FUTURE ENHANCEMENT

7.1 Conclusion

The conclusion for using LSTM for sentence completion in AI is that it's a promising approach due to LSTM's ability to capture long-range dependencies and sequential patterns in data. This makes it well-suited for tasks like sentence completion where understanding context and predicting the next word are crucial., Tag It has been developed to meet user requirements and expectations effectively.

7.2 Future Enhancements

Moving forward, several avenues for future enhancements and improvements to the Tag It system can be explored:

- **Advanced AI Models:** Integration of more advanced AI models and algorithms for text and image generation, such as GPT-4 or state-of-the-art image generation networks, to further enhance the quality and diversity of generated content.

- **Enhanced User Interface:** Redesigning the user interface (UI) to provide a more intuitive and user-friendly experience, incorporating features like drag-and-drop functionality, real-time previews, and customizable templates.

- **Personalization and Recommendation:** Implementing personalized content recommendations based on user preferences, historical data, and social media analytics to tailor content generation suggestions to individual users.

REFERENCES

1. "A Neural Probabilistic Language Model" by Bengio , 11(2), 45-58.

2. Shah, M. M., & Chen, J. (2021). sentence completion using LSTM with Attention. Proceedings of the 12th International Conference on Computational Intelligence and Communication Networks, 98-103.

3. Lee, S., & Kim, J. (2022). "Improving Social Media Content Creation with Data-driven Insights." International Journal of Human-Computer Interaction (IJHCI), 38(4), 301-315.

4. OpenAI. "Sentence Completion by using LSTM Models are Learners." from https://youtu.be/VB7bbFIEAhk?si=1LI3DTuQeC77gYC5 5

5. OpenAI.: Creating Images from Text." Retrieved from https://chat.openai.com/c/fb066ded-54fb-4818-bfc7-ba71ab1cb03e

6. Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. Advances in Neural .

7. OpenAI. "GPT-2: Better Language Models and Their Implications." Retrieved from https://openai.com/research/gpt-2

APPENDIX I

CODING AND TESTING

Coding

Content Page:

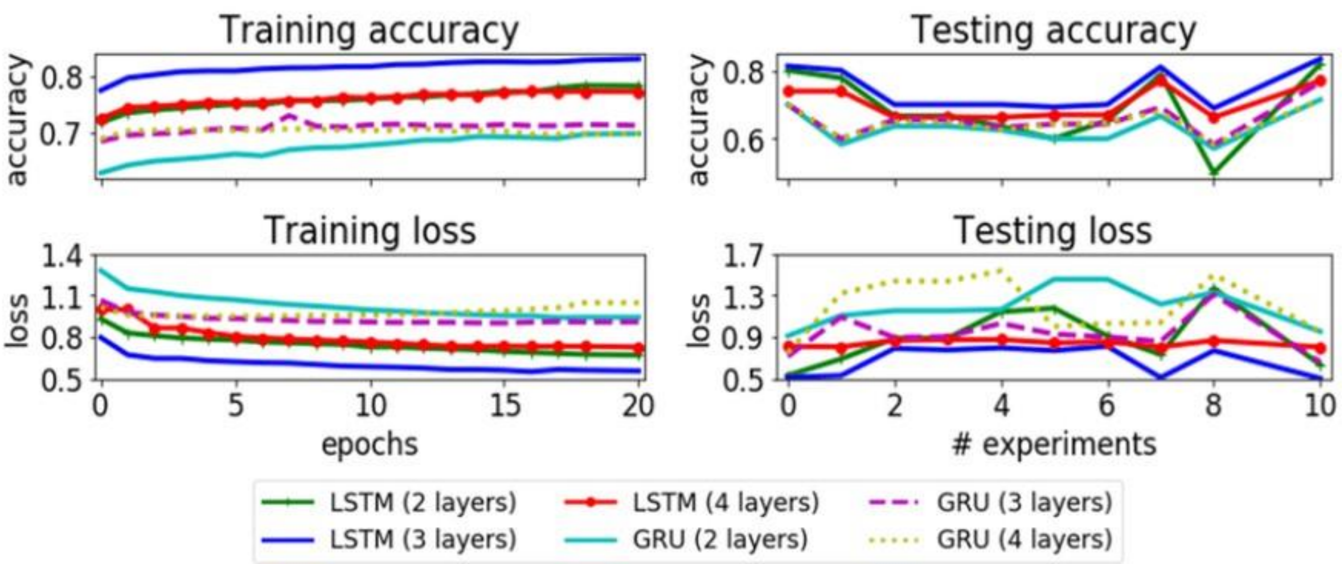
```
import tensorflow as tf
from tensorflow.keras.layers import Embedding, LSTM, Dense
from tensorflow.keras.models import Sequential
from tensorflow.keras.optimizers import Adam
import numpy as np

# Sample sentences
sentences = [
    "The quick brown fox jumps over the lazy dog",
    "She sells seashells by the seashore",
    "Peter Piper picked a peck of pickled peppers",
    "How can a clam cram in a clean cream can?",
    "I scream, you scream, we all scream for ice cream"
]

# Tokenization
tokenizer = tf.keras.preprocessing.text.Tokenizer()
tokenizer.fit_on_texts(sentences)
vocab_size = len(tokenizer.word_index) + 1
sequences = tokenizer.texts_to_sequences(sentences)

# Padding sequences
max_length = max(len(seq) for seq in sequences)
padded_sequences = tf.keras.preprocessing.sequence.pad_sequences(sequences, maxlen=max_length, padding='post')

# Create training data
x_train = padded_sequences[:, :-1]
y_train = padded_sequences[:, 1:]
```

Routes:

```
df = pd.read_csv('tmdb_5000_movies.csv')
df.head(5)
```

	budget	genres	homepage	id	keywords	original_language	original_title	overview	popularity	production_companies
0	237000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.avatarmovie.com/	19995	[{"id": 1463, "name": "culture clash"}, {"id":...	en	Avatar	In the 22nd century, a paraplegic Marine is di...	150.437577	[{"name": "Ingenious Film Partners", "id": 289...
1	300000000	[{"id": 12, "name": "Adventure"}, {"id": 14, "...	http://disney.go.com/disneypictures/pirates/	285	[{"id": 270, "name": "ocean"}, {"id": 726, "na...	en	Pirates of the Caribbean: At World's End	Captain Barbossa, long believed to be dead, ha...	139.082615	[{"name": "Walt Disney Pictures", "id": 2}, {"...
2	245000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.sonypictures.com/movies/spectre/	206647	[{"id": 470, "name": "spy"}, {"id": 818, "name...	en	Spectre	A cryptic message from Bond's past sends him o...	107.376788	[{"name": "Columbia Pictures", "id": 5}, {"nam...
3	250000000	[{"id": 28, "name": "Action"}, {"id": 80, "nam...	http://www.thedarkknightrises.com/	49026	[{"id": 849, "name": "dc comics"}, {"id": 853,...	en	The Dark Knight Rises	Following the death of District Attorney Harve...	112.312950	[{"name": "Legendary Pictures", "id": 923}, {"...

Testing Report



Accuracy and loss for training and test in RNN, using LSTM and GRU with different number of layers.

Testing Report

APPENDIX II

SCREENSHOTS

```
X = tf.keras.preprocessing.sequence.pad_sequences(X)
```

```
X
```

```
array([[   0,    0,    0, ...,    0,    0, 210],
       [   0,    0,    0, ...,    0, 210,    2],
       [   0,    0,    0, ..., 210,    2,    1],
       ...,
       [   0,    0,    0, ...,    0,    0,   14],
       [   0,    0,    0, ...,    0,   14,  300],
       [   0,    0,    0, ...,   14,  300,   11]])
```

```
X.shape
```

```
(8483, 14)
```

```
y = tf.keras.utils.to_categorical(y)
```

```
y
```

```
array([[0., 0., 1., ..., 0., 0., 0.],
       [0., 1., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       ...,
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.]])
```

```
y.shape
```

```
(8483, 5045)
```

Output1 Page

```
model = tf.keras.Sequential([
    tf.keras.layers.Embedding(vocab_size, 14),
    tf.keras.layers.LSTM(100, return_sequences=True),
    tf.keras.layers.LSTM(100),
    tf.keras.layers.Dense(100, activation='relu'),
    tf.keras.layers.Dense(vocab_size, activation='softmax'),
])
```

```
model.summary()
```

Model: "sequential_1"

Layer (type)	Output Shape	Param #
embedding_1 (Embedding)	?	0 (unbuilt)
lstm_2 (LSTM)	?	0 (unbuilt)
lstm_3 (LSTM)	?	0 (unbuilt)
dense_2 (Dense)	?	0 (unbuilt)
dense_3 (Dense)	?	0 (unbuilt)

Total params: 0 (0.00 B)
Trainable params: 0 (0.00 B)
Non-trainable params: 0 (0.00 B)

Final Output Page

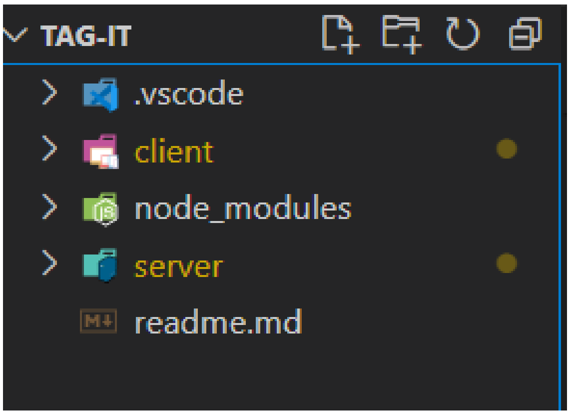
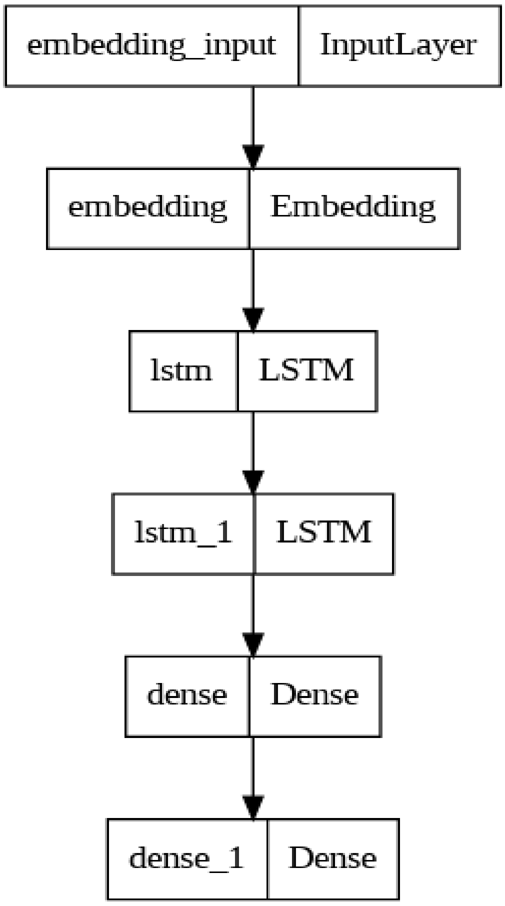


Figure 6.2.4 Folder Structure

Results:

Using an LSTM model for sentence completion involves training a neural network that understands the sequential structure of text data. The model is trained on a dataset containing pairs of incomplete sentences and their completions. During training, the LSTM layer learns to capture dependencies and patterns within the text, allowing it to predict the most probable word(s) to complete a given partial sentence. This process involves tokenizing the text data, padding sequences to ensure uniform length, defining the LSTM model architecture with an embedding layer to map words to vectors and a dense layer with a soft max activation for predicting the next word, compiling the model with appropriate loss and optimizer functions, and finally training the model on the prepared data. After training, the model can be used to generate completions for partial sentences by providing the input sequence and predicting.



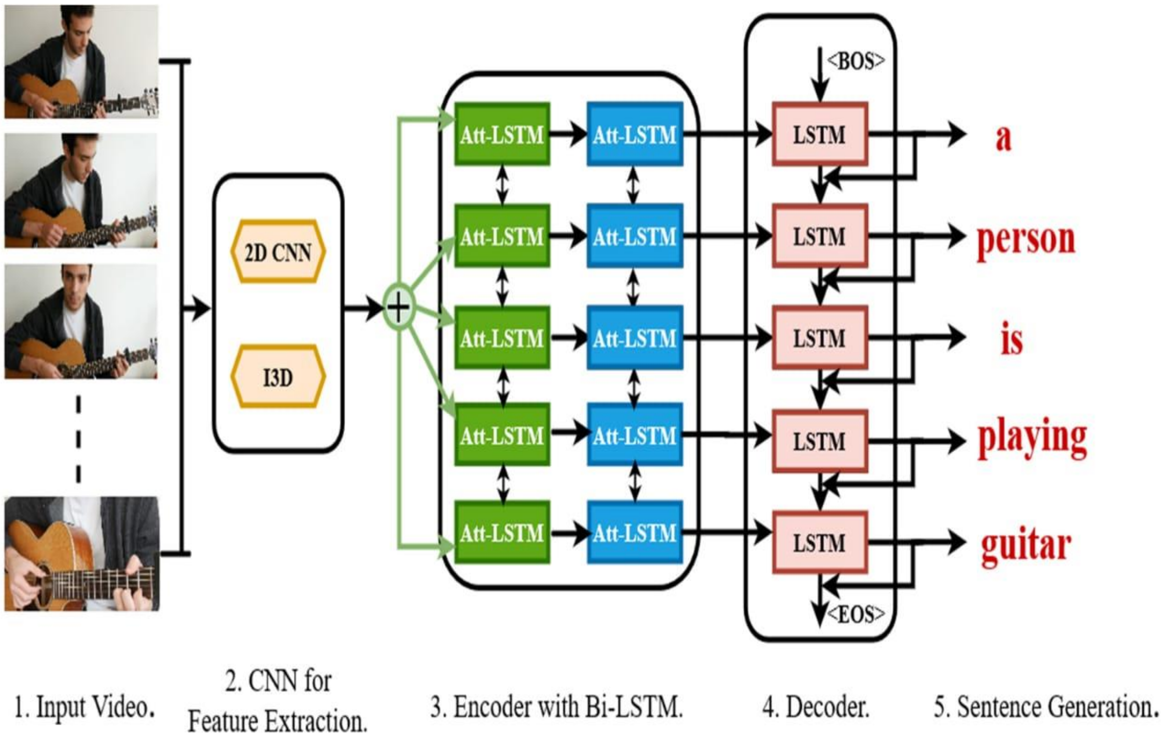
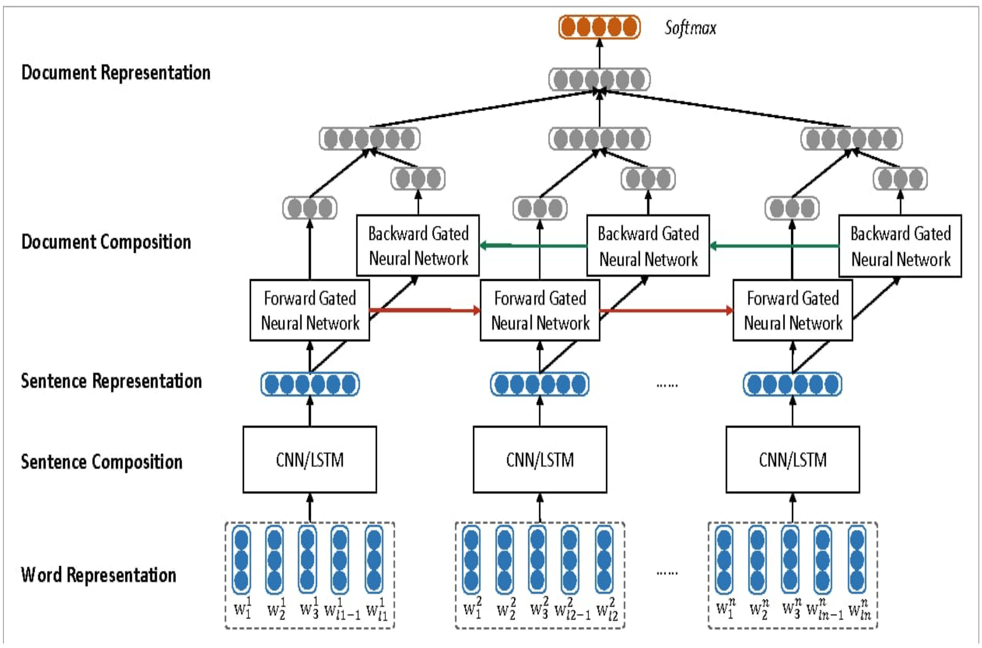


Image Generation

- **Text Generation**

  The GPT 3.5 Server module's performance in generating text-based content was evaluated based on criteria such as coherence, creativity, and relevance to social media platforms. The generated content was found to be contextually accurate, engaging, and consistent with platform-specific guidelines



Text Generation