Long Term Memory Chatbot

Aman Sharma MT24013 | Anupma Pandey MT23019 | Prajwal MT24064 | Rahul Gupta MT24070 | Sai Krishna Kota MT24078

1 Dataset Description

The dataset comprises multiple stories and corresponding questions and answers, formatted in JSON. Each entry contains a story, a set of questions based on the story, and their corresponding answers. This structure is ideal for tasks related to reading comprehension, chatbot development, or question-answering systems. The stories also have corresponding span texts associated with each question, from which the answer is to be extracted.

2 Exploratory Data Analysis

- Number of Stories: 500 stories in JSON format containing story answers with SpanText and additional answers for every question.
- Number of Questions per Story: 10-24 questions.
- Total Questions: 7983.
- Categories: 'What', 'Who', 'Which', 'How', 'Yes/No', 'When', 'Where', 'Why', 'Other'.
- NA Values for Answers: Every question has a corresponding answer; no NA values.

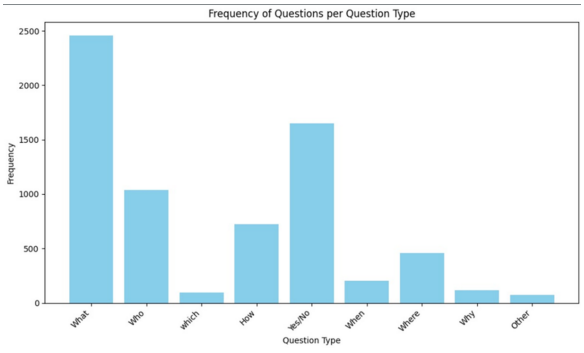

Figure 1: Frequency of questions per question type

In: W. Aigner, G. Schmiedl, K. Blumenstein, M. Zeppelzauer (eds.): Proceedings of the 9th Forum Media Technology 2016, St. Pölten, Austria, 24-11-2016, published at http://ceur-ws.org

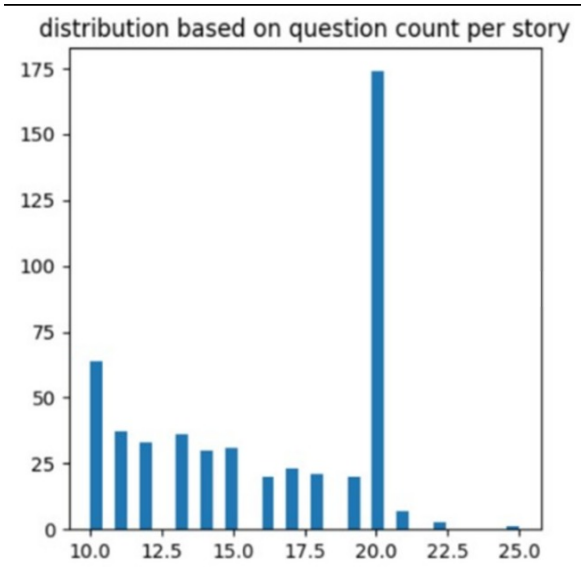

Figure 2: Distribution based on question count per story

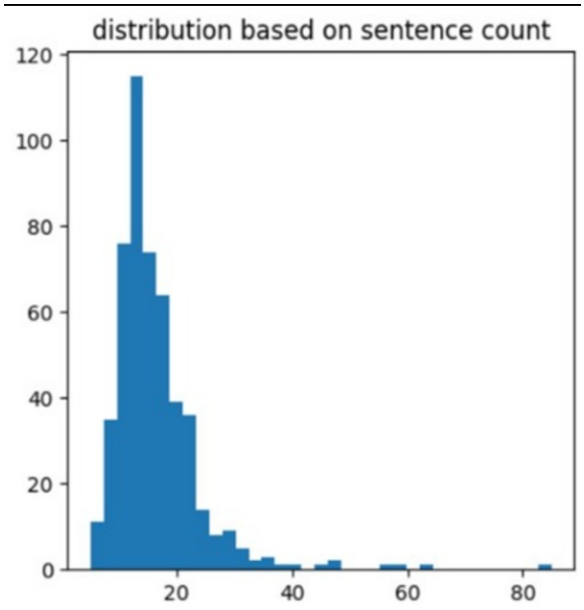

Figure 3: Distribution of sentence count over stories

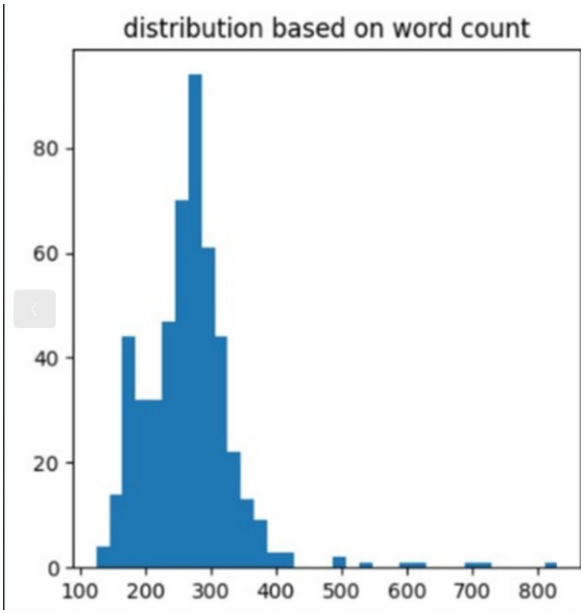

Figure 4: Distribution of sentence count over stories

3 Methodology

3.1 Data Preprocessing

- Removed irrelevant characters such as punctuation marks, special symbols, and extra spaces.
- Converted all text to lowercase to standardize the data and minimize case sensitivity issues.
- Split the text into individual words or tokens using the nltk library.
- Removed common stopwords (e.g., "and," "the," "is") using NLTK's predefined stopword list to reduce noise in the dataset.
- Reduced words to their base or root forms using WordNet lemmatizer (e.g., "running" to "run") to group similar words and improve consistency.

3.2 Story Identification

- Extracted named entities such as proper nouns, locations, and other key entities using NLTK's pos tag and ne_chunk functions.
- Aimed to identify key components for entity-based analysis.
- Identified synonyms of words using the WordNet lexical database.
- Expanded the dataset's semantic scope for enhanced query matching.
- Conducted frequency analysis to identify rare or unique words.
- Built word-frequency mappings to facilitate ranking and relevance determination.

3.3 Feature Engineering

- To enhance model performance, we categorized question-answer pairs into types such as "who," "what," "why," "how," "when," "where," and others.
- For each category, we extracted specific features like "has location context" for "where" questions and "has time context" for "when" questions.
- We also applied common features like word overlap (measuring shared words between the question and answer) and entity match (identifying matching named entities). This combination of category-specific and common features helped the model better understand the context and improve accuracy in selecting the correct answers.
- It then employs TF IDF vectorization and cosine similarity to rank the stories based on relevance to the input question. The most relevant story is retrieved and its sentences are displayed, offering a highly targeted and contextually appropriate response

3.4 Sentence Extraction

- In our project, answer extraction is performed by identifying and selecting relevant sentences from the retrieved story using its unique story id. After retrieving the story from the database, it is split into sentences. These sentences are analyzed to determine their relevance to the user's question. Relevance is assessed based on features like semantic similarity and contextual cues specific to the question type (e.g., "where," "when," "who"). The most relevant sentence is selected and return. This ensures that the response is given and directly addresses the user's query. In cases where no clear answer is found, fallback mechanisms are employed to provide a meaningful response or prompt the user for clarification.

3.5 PIPELINE

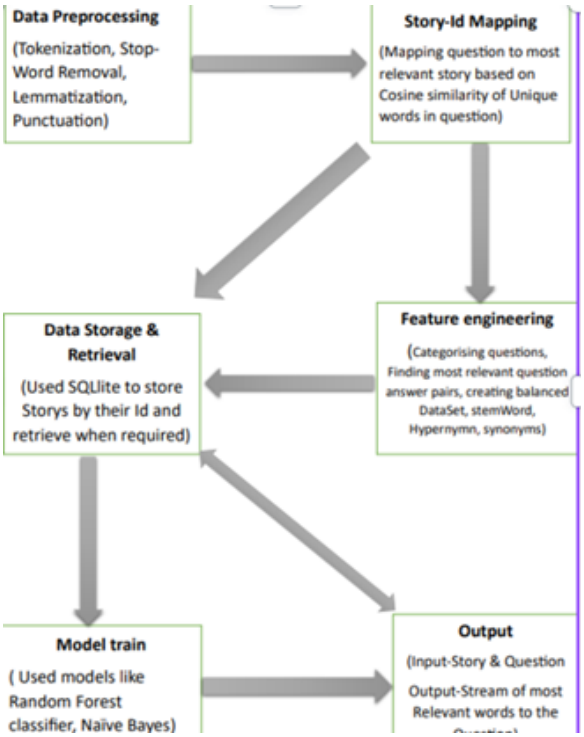

Figure 5: Figure 6

3.6 Models

We employed a Naive Bayes classifier to predict the relevance of question-answer pairs within the context of a story. The model operates on Bayes' theorem and assumes independence among features, making it computationally efficient and interpretable. The following features were utilized during training:

- Tailored attributes, such as "location context" for "where" questions or "temporal context" for "when" questions.
- Metrics like word overlap (shared vocabulary between questions and answers) and entity match (alignment of named entities such as names, places, or dates).

The Random Forest Classifier (RFC) was trained to classify user questions as valid or invalid based on features like semantic similarity and story context. This ensures only relevant questions are mapped to stories, enhancing system efficiency and response accuracy.

3.7 Results

- The Naive Bayes Classifier achieved an accuracy of 62.59 percentage, while the Random Forest Classifier had an accuracy of 60.57 Percentage. The Naive Bayes model outperformed the RFC, indicating its effectiveness for question validation within the given context. Both models showed moderate accuracy, suggesting room for improvement with further model tuning or feature enhancements.

Table 1: Model accuracy for answer prediction

Model	Accuracy
Naive Bayes	62.59
Random Forest	60.57

References

1 Answer Extraction in Question Answering using Structure Features and Dependency Principles, Lokesh Kumar Sharma, Namita Mittal, https://doi.org/10.48550/arXiv.1810.03918

2 CHATBOT: DESIGN, ARCHITECUTRE, AND APPLICATIONS, Xufei Huang

3 CoQA: A Conversational Question Answering Challenge, Siva Reddy, Danqi Chen, Christopher D. Manning, https://doi.org/10.48550/arXiv.1808.07042

4 A Smart Chatbot Architecture based NLP and Machine Learning for Health Care Assistance, Soufyane Ayanouz, et al, https://doi.org/10.1145/3386723.3387897

5 Comparison Of Multinomial Naive Bayes Algorithm And Logistic Regression For Intent Classification In Chatbot, Muhammad Yusril Helmi Setyawan; Rolly Maulana Awangga; Safif Rafi Efendi, DOI: 10.1109/INCAE.2018.8579372