

INTELLECTUAL PLAY WITH KIDS: BUILDING AN INTERACTIVE TOY BY USING LSTM MODEL**Dhanush S¹, Reema Ayeli², Monika K M³, Murli Raghavendra C S⁴ and Alpha Vijayan⁵**^{1,2,3,4,5} Department of CSE PES University Bangaluru, India¹pesug20cs428@pesu.pes.edu, ²pesug20cs452@pesu.pes.edu, ³pesug20cs439@pesu.pes.edu, ⁴pesug20cs440@pesu.pes.edu and ⁵alphavijayan@pesu.pes.edu

How to Cite: Dhanush, S., Ayeli, R., Monika, K. M., Murli Raghavendra, C. S. and Vijayan A. (2023), Intellectual Play with Kids: Building an Interactive toy by using LSTM Model. International Journal of Applied Engineering Research, International Journal of Applied Engineering & Technology X(X), pp.X-X.

ABSTRACT

Recent advancements in technology have significantly influenced intellectual play, notably through interactive toys fostering learning and development in children. These toys, powered by machine learning algorithms such as LSTM, offer tailored learning experiences, adapting to individual needs and interests. Unlike traditional toys with predefined words, these innovations allow for interactive and responsive interactions, mimicking a child's voice and employing machine learning to provide accurate and customized responses breakthrough in intellectual play provides children with a glimpse into the future through simple yet sophisticated toys. Leveraging machine learning, particularly LSTM, these toys understand queries and generate appropriate answers, offering endless entertainment and educational value. They serve as nurturing companions, capable of reciting a variety of rhymes and bedtime stories to comfort children. Their unique ability to continuously update and personalize content ensures a constant flow of new and innovative information, creating lifelike, real-time friendships with children. Ultimately, this project's goal is to integrate machine learning into children's education and communication development, offering an interactive and engaging learning platform that simultaneously entertains and enhances productivity.

Index Terms: LSTM, Deep Learning Model, NLP, Voice Recognition.

INTRODUCTION

Intellectual play with kids is an approach to play that emphasizes the development of cognitive, social, and emotional skills through interactive and engaging activities. This type of play can take many forms like playing rhymes, bedtime stories, etc. In recent years, technology has played an increasing role in intellectual play. Interactive toys have provided new and innovative ways to engage children in learning and development. Machine learning algorithms like LSTM have made it possible to create intelligent and adaptive toys that can customize the learning experience to suit a child's specific needs and interests. In conclusion, intellectual play with kids is an essential approach to play that fosters cognitive, social, and emotional development. It has numerous benefits for children's learning and growth and can be used to create a positive and engaging learning environment. The integration of technology and machine learning algorithms like LSTM can provide new opportunities for interactive and adaptive play, enabling children to learn and grow in an exciting and innovative way.

LITERATURE SURVEY

Paper 1] To enhance the model's capacity for comprehending and responding to user inquiries, the authors suggest a chatbot model that makes use of a Bidirectional Recurrent Neural Network (BRNN) with an attention mechanism. The attention mechanism directs the model's attention to particular segments of the input sequence, which helps the model comprehend the context and provide more accurate replies. The chatbot model combines a combination of supervised and unsupervised learning approaches to extract knowledge from a huge collection of conversation transcripts during training. The model's performance is assessed by the authors using several measures, including accuracy, perplexity, and F1 score, and it is then contrasted with other cutting-edge chatbot models. Overall, the study demonstrates an intriguing use of deep learning techniques to create a chatbot that is

International Journal of Applied Engineering & Technology

intelligent and can have natural language interactions with humans. The paper's findings may aid in advancing the creation of future chatbots that are more effective and complex because of the model's capacity to comprehend the context and produce accurate responses.

Paper[2]The authors begin by outlining the rising significance of automated text generation, which has several uses in industries including artificial intelligence, machine learning, and natural language processing. They also draw attention to the difficulties in creating efficient automated text production models, including the necessity for big, varied datasets and the complexity of human language. The paper then provides a thorough analysis of various deep learning approaches, including Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, and Gated Recurrent Units (GRUs), utilized in automated text synthesis. The authors go over each technique's advantages and disadvantages as well as how to use it in various situations. The authors give a case study on the creation of news headlines using a combination of RNNs and LSTM networks to show the efficacy of these techniques and evaluation measures. Using the various assessment criteria previously stated, they assess the model's performance and demonstrate that their model exceeds other current models in terms of correctness and coherence. Overall, the study paper offers a thorough examination of deep learning-based automated text production. The authors stress the significance of creating efficient automated text production models and offer insights into the many methods and assessment measures that can be employed to accomplish this objective. Overall, the study paper offers a thorough examination of deep learning-based automated text production. algorithms or retrieval-based techniques that don't generate good results. In this paper, they compared the performance of three chatbots built by using RNN, GRU, and LSTM.

Paper[3]The use of speech-to-text conversion and summarization techniques for efficient understanding and documentation is covered in this research paper. The authors emphasize the value of efficient documentation in a variety of industries, including business, healthcare, and education, as well as the difficulties of using conventional documentation techniques. To address these issues, they present speech-to-text conversion and summarising algorithms, which can enhance the precision and effectiveness of documentation. The technical challenges of putting speech-to-text conversion and summarization systems into practice are covered in the study, including the application of Natural Language Processing (NLP) methods and machine learning algorithms. The authors provide a thorough overview of the many methods for speech-to-text conversion, including models based on deep learning, hidden Markov models, and Gaussian mixture models. The many methods for text summarization, including extractive and abstractive summarization, are also covered in the study, along with the difficulties that each method presents. The case study the authors give illustrates the viability of the suggested strategy by implementing a speech-to-text conversion and summary system for the healthcare industry. The entire use of speech-to-text conversion and summary approaches for efficient understanding and documenting is covered in this study. The study may contribute to the advancement of more successful and sophisticated speech-to-text conversion and summarization systems, with ramifications for many fields that demand accurate and effective documentation.

Paper[4]The greater diversity in children's speech patterns and the scarcity of annotated data, according to scientists, make ASR for children more difficult than ASR for adults. They suggest a unique method that directly learns features from unprocessed speech signals using deep learning techniques, which can enhance the effectiveness of ASR systems for young users. The proposed method, which involves teaching a convolutional neural network (CNN) on unprocessed voice data to learn distinguishing features, is presented in full in the study. The learned traits are subsequently utilized to train a speech recognition system, which can more accurately identify children's speech. The authors run trials on a publicly accessible dataset of children's speech to show the efficacy of their methodology. They demonstrate how their methodology beats other established feature extraction methods like Mel-frequency cepstral coefficients (MFCCs) by comparing their accuracy to those methods. The report also examines the approach's shortcomings and offers research directions for the future. The authors draw attention to the need for additional research on how learned features might be used to other speech tasks and how to adapt the suggested methodology to other languages. Overall, the study offers a novel strategy for enhancing

ASR in kids that could have significant repercussions for several applications, including speech therapy and educational technology. The discoveries might potentially help deep learning methods for speech processing and recognition advance.

Paper[5]The research paper "Design and Implementation of Text to Speech Synthesiser using Syllabification Synthesis Algorithm and Comparing with Articulator Synthesis Algorithm" conducts a comparison of the syllabification synthesis algorithm and the articulator synthesis algorithm, two different methods for text-to-speech (TTS) synthesis. The authors of the report emphasize the significance of TTS synthesis as a field of study with numerous applications, such as assistive technology, tools for language acquisition, and automated voice response systems. Their research's objective is to contrast the two distinct TTS synthesis algorithms and assess each one's performance using different metrics. The syllabification and articulator synthesis algorithms, as well as their underlying theories and the processes necessary for the synthesis process, are thoroughly described in this work. The authors also go into detail on how the two methods were put into practice using various tools and programming languages. The choice of the TTS synthesis algorithm, according to the authors, should be based on the requirements of the individual applications and the trade-off between accuracy, naturalness, and processing speed. Overall, the study paper presents a thorough analysis of two distinct methods for TTS synthesis and contrasts their performance according to several factors. The results may be helpful for academics and industry professionals involved in TTS synthesis, and they may aid in the creation of TTS systems that are more effective and precise.

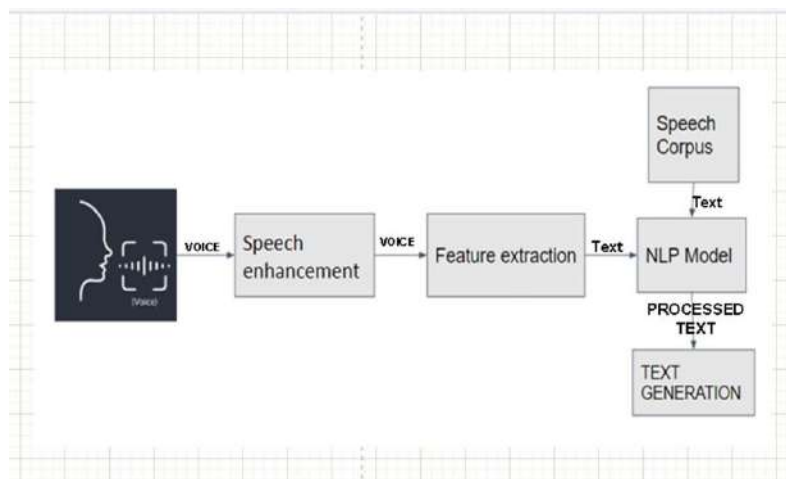


Fig. 1: Speech-To-Text

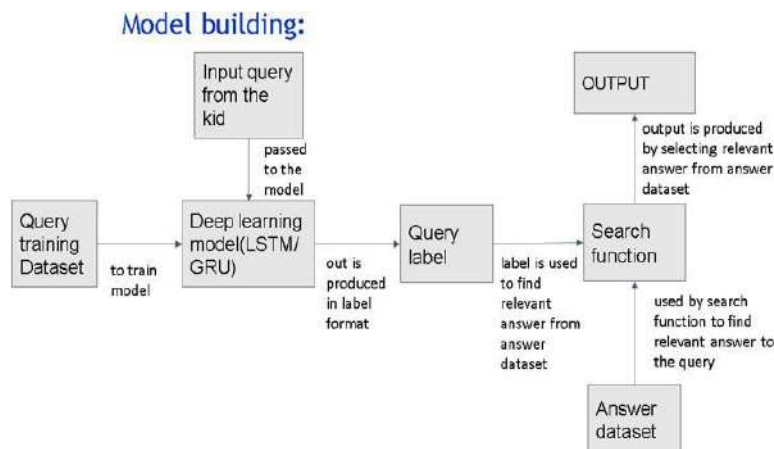


Fig. 2: Model Building

METHODOLOGY

A. Speech to Text Module

- Converting speech to text: Natural language processing can be used for feature extraction, Natural language toolkit, and Speech recognition can be used to convert speech to text.
- Speech Recognition Component: This component is re- sponsible for recognizing and interpreting the child’s spoken input
- Google API: This is used to convert detected speech into text format
- Natural Language Processing (NLP) Component: the NLP is used to generate segmented text from the input text (**refer fig 1**)

B. Model Building

Model building: A deep learning model is used which is trained on a dataset with queries and labels. The model recognizes the input query and gives a label. Output is provided based on the label generated. (**refer fig 2**)

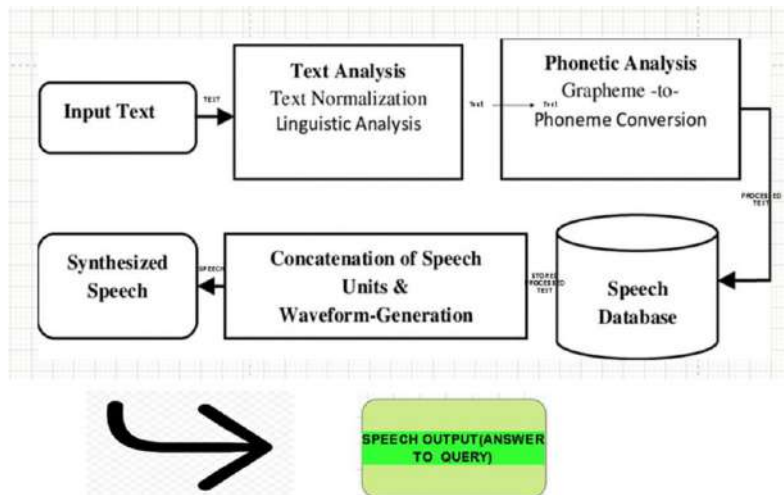


Fig. 3: Text To Speech Module

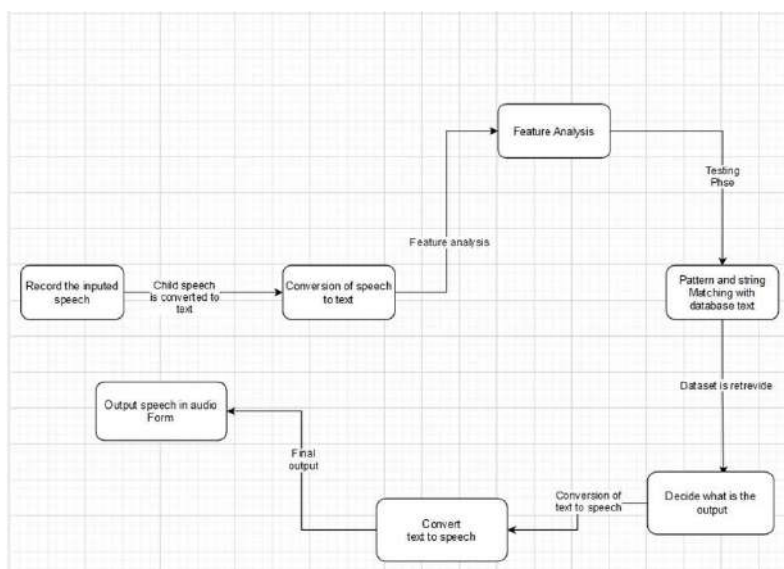


Fig. 4: Integrated System

C. Text to Speech Module

Converting text to speech: The output is syllabified and broken into various parts and afterward text investigation is done to combine the content into speech.(refer fig 3)

D. Integrated System

The system is integrated by combining all three modules. The output of speech to text module is passed to the processing module. The processing module will produce appropriate answers in the form of text. If the answer is playing rhymes or telling stories, they are played directly from the storage. The text output is passed through text to text-to-speech conversion module to produce speech output. All this flow is represented in fig 4 .(refer fig 4,5)

DATA SET

To complete the implementation of our project we have used two different datasets. They are Rhymes and stories dataset: This dataset contains numerous rhymes and stories in mp3 format which are stored in the storage of the Raspberry Pi model and played on the request of the kid/user.

- **Conversational Dataset:** This dataset contains all possible conversations that can be made between the kids. This dataset is used to understand the query given by the kid and generate appropriate replay for the kid/user. This dataset is used to train and test the LSTM model built for conversation purposes.

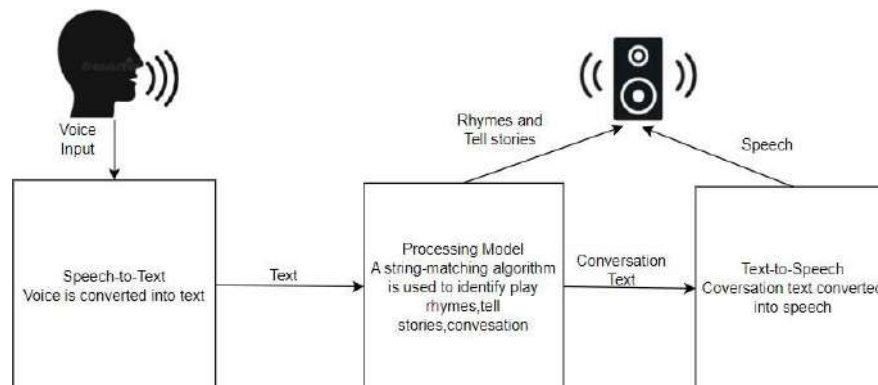


Fig. 5: Model Architecture

```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS Python + [
ALSA lib pcm_usb_stream.c:482:(snd_pcm_usb_stream_open) Invalid card 'card'
ALSA lib pcm_dmix.c:999:(snd_pcm_dmix_open) unable to open slave
Cannot connect to server socket err = No such file or directory
Cannot connect to server request channel
jack server is not running or cannot be started
JackShmReadWritePtr::~JackShmReadWritePtr - Init not done for -1, skipping unlock
JackShmReadWritePtr::~JackShmReadWritePtr - Init not done for -1, skipping unlock
You said: hello
  
```

Fig. 6: Detection Of Wake Word

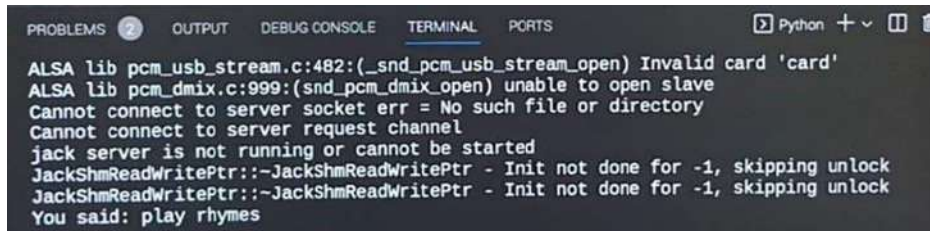
RESULTS

A. Speech to Text Module

- The module uses a Speech recognition library which allows the system to detect the speech query from the kid/user.
- Google API is used to generate the text format of the kids' query.
- Using above both techniques the speech from the kid is detected and converted into text format successfully.
- The generated text is passed to the processing module.
- The refer fig:6, 7, 8 displays the text detected and converted by speech to text module.

B. Processing Module

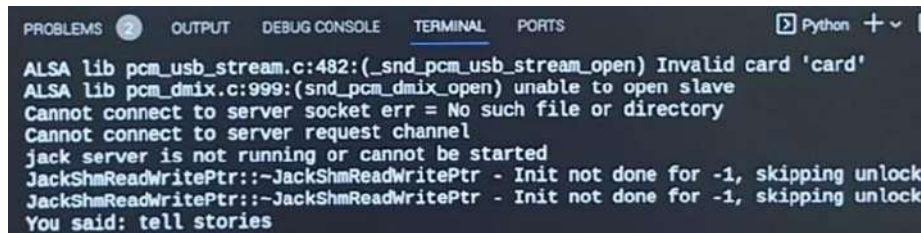
- The processing module is built by the combination of a string-matching algorithm and deep learning model.



```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS Python + v
ALSA lib pcm_usb_stream.c:482:(snd_pcm_usb_stream_open) Invalid card 'card'
ALSA lib pcm_dmix.c:999:(snd_pcm_dmix_open) unable to open slave
Cannot connect to server socket err = No such file or directory
Cannot connect to server request channel
jack server is not running or cannot be started
JackShmReadWritePtr::~JackShmReadWritePtr - Init not done for -1, skipping unlock
JackShmReadWritePtr::~JackShmReadWritePtr - Init not done for -1, skipping unlock
You said: play rhymes
  
```

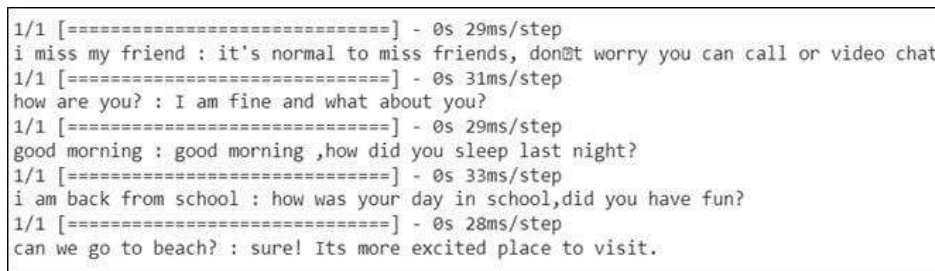
Fig. 7: Detection of “Play Rhymes” Command



```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS Python + v
ALSA lib pcm_usb_stream.c:482:(snd_pcm_usb_stream_open) Invalid card 'card'
ALSA lib pcm_dmix.c:999:(snd_pcm_dmix_open) unable to open slave
Cannot connect to server socket err = No such file or directory
Cannot connect to server request channel
jack server is not running or cannot be started
JackShmReadWritePtr::~JackShmReadWritePtr - Init not done for -1, skipping unlock
JackShmReadWritePtr::~JackShmReadWritePtr - Init not done for -1, skipping unlock
You said: tell stories
  
```

Fig. 8: Detection of “Tell Stories” Command



```

1/1 [=====] - 0s 29ms/step
i miss my friend : it's normal to miss friends, don't worry you can call or video chat
1/1 [=====] - 0s 31ms/step
how are you? : I am fine and what about you?
1/1 [=====] - 0s 29ms/step
good morning : good morning ,how did you sleep last night?
1/1 [=====] - 0s 33ms/step
i am back from school : how was your day in school,did you have fun?
1/1 [=====] - 0s 28ms/step
can we go to beach? : sure! Its more excited place to visit.
  
```

Fig. 9: Results Obtained From Lstm Model

- Using string matching algorithm “hello” is detected to wake the processing module.
- “Play rhymes” and “tell stories” are the inbuilt commands to play rhymes and tell stories using a string-matching algorithm.
- conversation system produces accurate output for kids’ queries.
- The Conversation system is built by training the LSTM model. The following details the hyperparameters used to train the model: LSTM model with 128 cells, relu and softmax activation functions, sparse categorical cross- entropy loss function, and adam optimizer.
- After evaluating the LSTM model for the test data shown in **fig 9,10** the following results are obtained and study of LSTM model.
- In **fig 11** shows the variation in accuracy and loss during model testing. The increasing curve of accuracy and decreasing curve of loss can be observed, which is a good sign that the model is doing its job well.
- In **fig 12**, precision-recall curves of many classes align with each other indicating that the model’s performance is less dependent on the specific choice of the classification threshold. This could be beneficial when finding an optimal threshold is challenging or not critical.
- The model demonstrates a consistent ability to correctly identify instances of each class while minimizing false positive and false negative values.

C. Text to Speech Module

The text output from the conversation is converted into audio output by using gTTs library and pygame library.

Model	Training accuracy	Training loss	Testing accuracy	Testing loss	Precision score
LSTM model	0.9488	0.1282	0.9731	0.1168	0.9523

Fig. 10: Study of Lstm Model

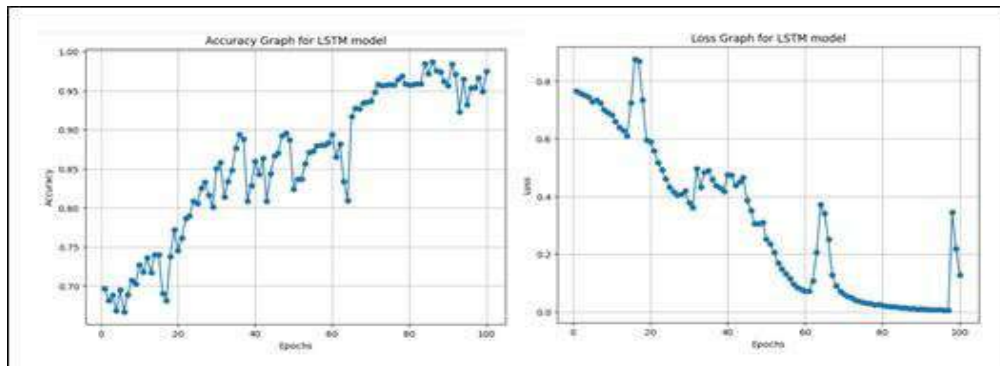


Fig. 11: Accuracy and Loss Graphs of Lstm

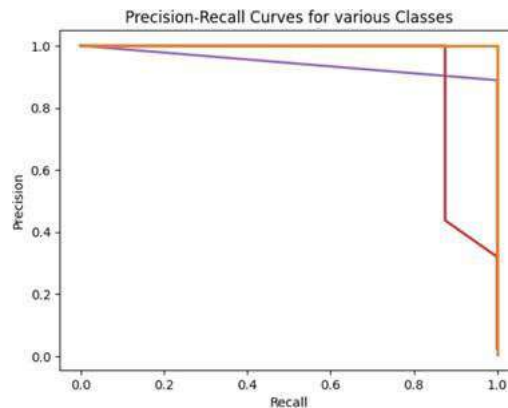


Fig. 12: Precision-Recall Curve of Lstm Model

- The output is passed through speakers which are connected to Raspberry Pi board.

D. Integrated System

- “HELLO” is used as a wake word for the toy. When the user/kid says the wake word, it activates the toy and enables it to listen for queries from the user/kid.
- The speech-to-text module listens to a user query using a speech recognition library and converts it into text format with the help of Google API.
- If the text matches the command “play rhymes” the processing module will play random rhyme from the collection or if it matches the command “tell stories” the Processing module starts telling any one story from its collection.
- Rhymes or stories can be heard through speakers.
- The toy will understand any other query from the user/kid using the deep learning LSTM model present in the processing module and generate appropriate output for the query in text format.

International Journal of Applied Engineering & Technology

- This text output is passed to the text-to-speech module where speech output is generated and played through the speakers.

CONCLUSION

In this project, we are using speech recognition and deep learning model LSTM, string matching algorithms, and natural language processing techniques to build a system that is helpful for kids' moral, educational, and communicational growth. Robotic technology can be found in many areas like stores, hospitals, homes, restaurants, and workplaces. But this intellectual play with kids is a breakthrough where kids as a younger generation get to see the future world through simple little toys.

- The toy successfully detects the input query asked by the kid and converts it into text format.
- Our system enables the kid/user to ask different queries to the toy and the toy will generate an appropriate answer to the query proposed by the kid/user.
- It also enables a feature where the rhymes and stories are played at the request of the kid or user to provide entertainment to the kid which helps to improve moral growth and thinking in children.
- Our project is built to entertain and educate the children. The toy helps increase the creative thinking of children.

FUTURE WORK

- This project can be collaborated with other projects like object detection, emotion detection, etc. to create a child security program.
- Many other methods can be used other than LSTM to make the system more reliable and efficient.

ACRONYMS AND ABBREVIATIONS

- RNN: Recurrent Neural Network
- LSTM: Long Short Time Memory.
- API: Application Programming Interface.
- LLD: Low Level Design

REFERENCES

- [1] An intelligent Chatbot using deep learning with Bidirectional RNN and attention model Manyu Dhyani, Rajiv Kumar G. L. Bajaj Institute of Technology and Management, Greater Noida, Uttar Pradesh, India .16 May 2020. DOI:10.1016/j.matpr.2020.05.450..
- [2] Analysis of Automated text generation using Deep learning. Ankit Kumar, Abhishek Singh, Arnav Kumar, Dr. Manoj Kumar Dept. of Computer Science and Engineering Delhi Technological University New Delhi, India. 2021 International Conference on Computational Intelligence and Communication Technologies (CCICT).2021 IEEE —DOI: 10.1109/CCICT53244.2021.00014
- [3] Speech-to-text conversion and summarization for effective understanding and documentation Vinnarasu A., Deepa V. Jose Department of Computer Science, Christ (Deemed to be University), India. International Journal of Electrical and Computer Engineering (IJECE) Vol. 9, No. 5, October 2019
- [4] Improving children speech recognition through feature learning from raw speech signal S. Pavankumar Dubagunta, Selen Hande Kabil, and Mathew Magimai.Doss Idiap Research Institute, Martigny, Switzerland Ecole polytechnique fédérale de Lausanne (EPFL), Switzerland 2019
- [5] Design and Implementation of Text to speech synthesizer using Syllabification synthesis algorithm and comparing with Articulator Synthesis algorithm. E Alexandra, Dr.P Shyamala Bharathi Dept. of Electronics and Communication Engineering, Saveetha University, Chennai, India, 2022 International Conference on Business Analytics for Technology and Security (ICBATS)