

## Abstract

A chatbot is a software which is capable of communicating with human by using natural language processing. In our project, we plan to develop a python-based chatbot that integrates theory of computation (TOC) concepts, including finite automata and regular expressions. The chatbot interact with users, recognizing patterns and keywords in their inputs to answer their questions related to TOC.

## Introduction

Our project takes inspiration from the history of chatbots, including key moments like Alan Turing's Turing Test and the development of chatbots like Eliza and SmarterChild. We trained the chatbot using the basics of computation theory (TOC) with Python. This chatbot can understand how people talk and respond accordingly.

## Methodology

The concepts used in chatbot:

1. Pattern Matching
2. Algorithm: Multinational Naïve Bayes
3. Artificial Neural Network

### Components in Chatbot Architecture:

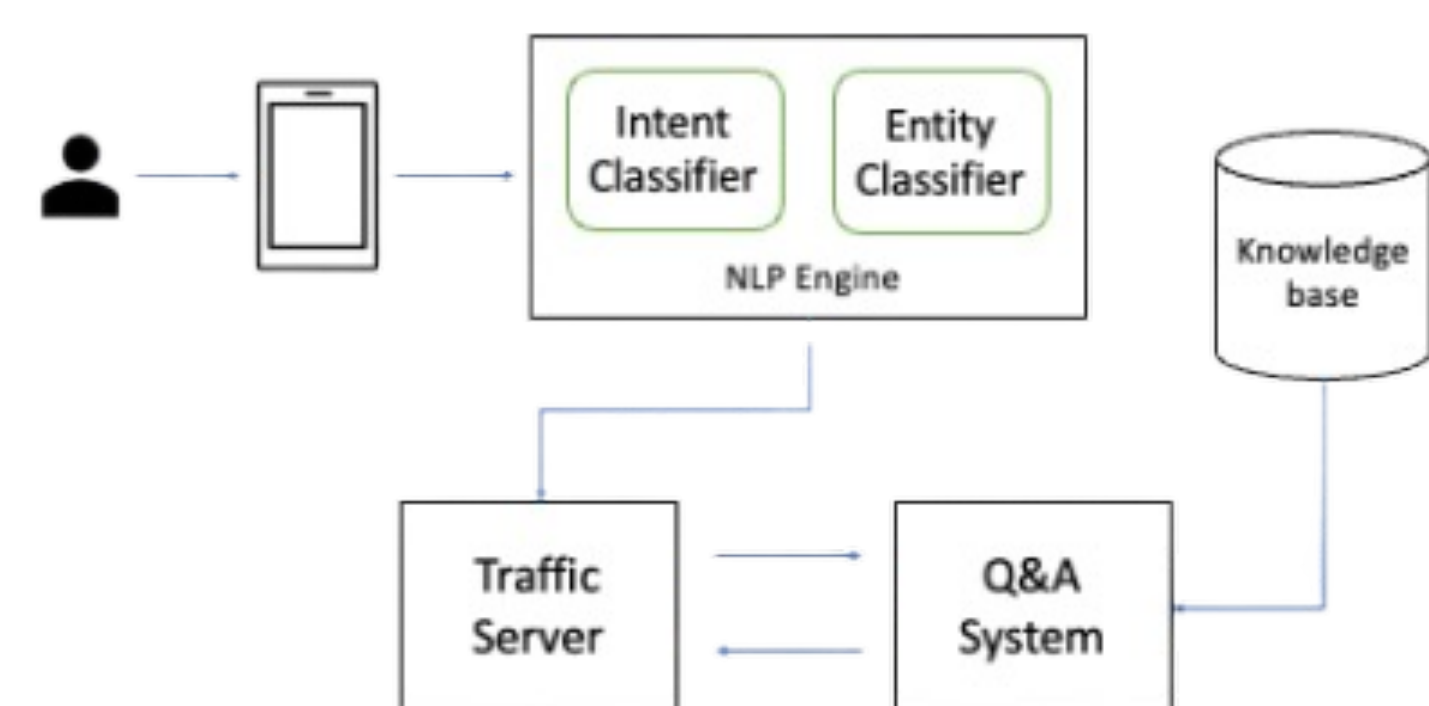


Fig 1: The architecture of chatbot.[4]

- Front End Systems:** Interaction with clients via website or various mobile apps.
- Environment:** Contextualizes users' messages by decoding and converting them into structured inputs.
- Traffic Server:** Handles user traffic requests, directs them to the appropriate components.
- Q/A System:** Analyzes the query and provide suitable answers from the knowledgebase, created either manually or via automated training.
- Custom Integrations:** Seamless integration with existing backend systems, allowing effortless linkage with critical components such as CRM, databases, payment applications, calendars, and other tools.

## Results

What is the difference between DFA and NFA?

**A Deterministic Finite Automaton (DFA) has a unique transition for each state and input symbol, while a Nondeterministic Finite Automaton (NFA) allows multiple possible transitions for a given state and input symbol.**

## Experiments

### Chatbot Training Workflow:

**Data Preparation:** At first, we loaded the PDF format of the textbook [5] using the Python PyPDF2 library [ 3]. Then, the loaded document is split into smaller chunks, and each chunk is processed to create embeddings. These embeddings represent the content in numerical form, making it easier for the chatbot to understand and search through the data.

**Data Storage:** After preparing the dataset to be understandable for the chatbot, we indexed the embeddings, as well as the document chunks, for efficient data retrieval. In our code case, the data is stored in ChromaDB[1], which is an in-memory storage solution for quick access.

**Query Processing:** When a user submits a query or question to the chatbot, the query is first embedded using the same embedding model used during data preprocessing. Then, the chatbot performs a semantic search through the indexed data and embeddings to extract relevant information related to the user's query. This process ensures that the chatbot can provide contextually accurate responses Language Model: Users' query and the relevant data passed to an LLM (Language Model).

**Language Model:** The chatbot then passes both the user's query and the relevant data to an LLM (Language Model). We used the "bigscience/bloom-1b7" language model provided by the HuggingFace[2] library. This model is responsible for generating responses based on the input data and query.

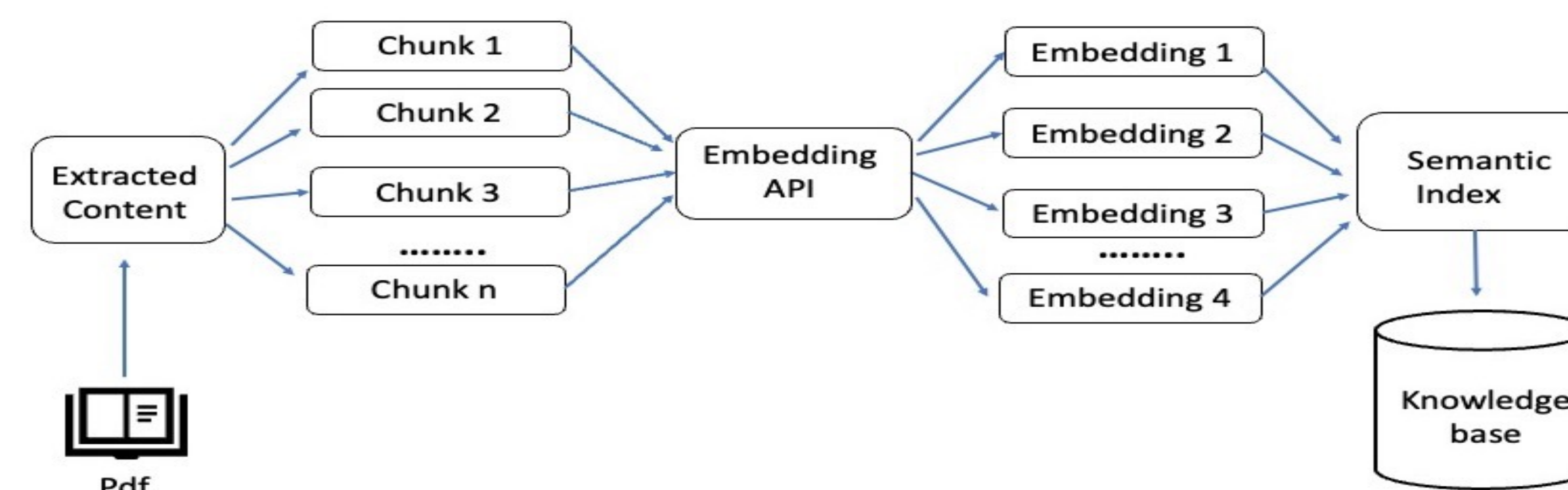


Fig 2: The workflow of chatbot training process. [7].

## Conclusions

- A python-based chatbot for theory of computation successfully trained using a textbook.
- Interactive interface making the chatbot accessible and practical.
- Represents the intersection of theory and practicality, bridging automata theory with real-world applications.
- Aims to unlock the chatbot's full potential and offer a valuable resource for those exploring the theory of computation.

## Contact Information

1. Sumaiya Tasneem, stasneem@students.Kennesaw.edu
2. Chinni Cherrishma Reddy Aduri, Caduri@students.Kennesaw.edu
3. Krishna Vamsi Anche, kanche@students.Kennesaw.edu
4. PURNA PAVAN KUMAR KOLLI, pkolli2 @students.Kennesaw.edu
5. Sharon Elugoti, selugoti@students.kennesaw.edu

## References

- [1] Chroma. <https://docs.trychroma.com/>
- [2] Hugging Face - Documentation. <https://huggingface.co/docs>
- [3] Welcome to PyPDF2 - PyPDF2 documentation. <https://pypdf2.readthedocs.io/en/3.0.0/>
- [4] Mirant Hingrajia. Conversational AI chatbot: Architecture Overview - QBurst blog. <https://blog.qburst.com/2020/09/conversational-ai-chatbot-architecture-overview/>
- [5] John E. Hopcroft, Rajeev Motwani, and Jeffrey D. Ullman. 2022. Introduction to automata theory, languages, and computation. Pearson Education.
- [6] Lorenz Cuno Klopfenstein, Saverio Delpriori, Silvia Malatini, and Alessandro Bogliolo. 2017. The rise of bots: A survey of conversational interfaces, patterns, and paradigms. In Proceedings of the 2017 conference on designing interactive systems.
- [7] Sudarshan Koirala. 2023. How to build a chatbot to chat with your PDF. (Aug 2023). <https://blog.gopenai.com/how-to-build-a-chatbot-to-chat-with-your-pdf-9abb9beaf0c4>
- [8] Pavel Kucherbaev, Alessandro Bozzon, and Geert-Jan Houben. 2018. Human aided bots. IEEE Internet Computing 22, 6 (2018), 36–43.
- [10] Maria das Graças Bruno Marietto, Rafael Varago de Aguiar, Gislene de Oliveira Barbosa, Wagner Tanaka Botelho, Edson Pimentel, Robson dos Santos França, and Vera Lúcia da Silva. 2013. Artificial intelligence markup language: a brief tutorial. arXiv preprint arXiv:1307.3091 (2013).
- [11] György Molnár and Zoltán Szűts. 2018. The role of chatbots in formal education. In 2018 IEEE 16th International Symposium on Intelligent Systems and Informatics (SISY). IEEE, 000197–000202.
- [12] Ketakee Nimavat and Tushar Champaneria. 2017. Chatbots: An overview types, architecture, tools and future possibilities. Int. J. Sci. Res. Dev 5, 7 (2017), 1019–1024.
- [13] Eleni Adamopoulou and Lefteris Moussiades. 2020. An Overview of Chatbot Technology. In Artificial Intelligence Applications and Innovations, Ilias Maglogiannis, Lazaros Iliadis, and Elias Pimenidis (Eds.). Springer International Publishing, Cham, 373–383
- [14] Soufyane Ayanouz, Boudhir Anouar Abdelhakim, and Mohammed Benhmed. 2020. A Smart Chatbot Architecture Based NLP and Machine Learning for Health Care Assistance. In Proceedings of the 3rd International Conference on Networking, Information Systems & Security (NISS2020). Association for Computing Machinery, New York, NY, USA.