LLM's for Societal Good - AAC Use Case

Mohan Kakarla
50596142 / mkakarla
mkakarla@buffalo.edu

Udit Brahmadevara
50608791 / uditbrah
uditbrah@buffalo.edu

Abstract

Individuals with speech impairments usually use Augmentative and Alternative Communication (AAC) systems to facilitate communication. However, current AAC tools often produce generic and impersonal outputs, failing to capture the nuances of individual personality, experiences, and intended conversational intent. These limitations results in reduced expressiveness, diminished user engagement, and inefficiencies in AAC-mediated interactions, ultimately hindering the full communicative potential of these individuals. Consequently, there is a critical need to develop and implement intelligent AAC systems that leverage the capabilities of Large Language Models (LLMs) to generate personalized and contextually relevant dialogues, thereby fostering more authentic and effective communication.

This project aims to address this gap by exploring the integration of Retrieval-Augmented Generation (RAG) and personalized prompting strategies to create an LLM-driven AAC system that better reflects the unique communication styles and personal narratives of its individual users, with the underlying goal of enhancing their social inclusion and quality of life.

1 Introduction

Augmentative and Alternative Communication (AAC) systems play a vital role in enabling individuals with speech impairments to communicate effectively. However, current AAC tools often fall short in providing truly personalized and expressive communication, producing outputs that can be generic and fail to capture the individual's unique personality, experiences, and intended conversational nuances. This limitation can lead to reduced expressiveness, lower user engagement, and inefficient use of AAC-mediated interactions, ultimately hindering the full communicative potential of AAC users and negatively impacting their social inclusion and overall quality of life.

To address these challenges, there is a need for intelligent AAC systems that leverage the power of Large Language Models (LLMs) to generate more personalized and contextually relevant dialogues. This project explores the integration of Retrieval-Augmented Generation (RAG) and personalized prompting strategies to develop an LLM-driven AAC system that aims to better reflect the unique communication styles and personal narratives of individual users, with the overarching goal of enhancing their social inclusion and quality of life. For this project, we introduce to types of users. The user that uses the AAC system is called the AAC communicator and the person interacting with the AAC communicator is called the primary user.

2 Model

For this project, there were many off the shelf LLM models available for fine tuning for our particular use case. But we went for the Llama-3-8B-Instruct-bnb-4bit model, which is built upon the LLama 3 architecture. It is a model that is known for its strong language-generation tasks. This particular model has been fine tuned on instruction following datasets.

The model has 8 billion parameters which strike a balance between performance and computational efficiency. It is large enough to solve complex problems but simple enough to be trained on a simple GPU. The model's weights have been quantized to 4-bit precision using the bitsandbytes library allowing it to limit RAM consumption.

We also used the all-MiniLM-L6-v2 model which is a pretrained sentence embedding model developed by SentenceTransformer. It is a lightweight model that is particularly used for efficient semantic similarity tasks.

3 Dataset

As the purpose of this LLM is to communicate with other users, the LLm should be able to provide

Feature	EmpatheticDialogues	GoEmotions
Dataset Type	Dialogue dataset	Reddit comments dataset
Data Source	Crowdsourced conversations on emotional prompts	Reddit comments
Context Length	Multi-turn dialogues	Single utterances
Emotion Focus	Emotionally grounded dialogue responses	Primary focus on emotion classification
Emotion Categories	32 categories (e.g., grateful, anxious, joyful)	27 categories (e.g., admiration, amusement, annoyance)
Data Size	(e.g., 25k dialogues)	(e.g., 58k comments)

Table 1: Comparison of the EmpatheticDialogues and GoEmotions datasets.

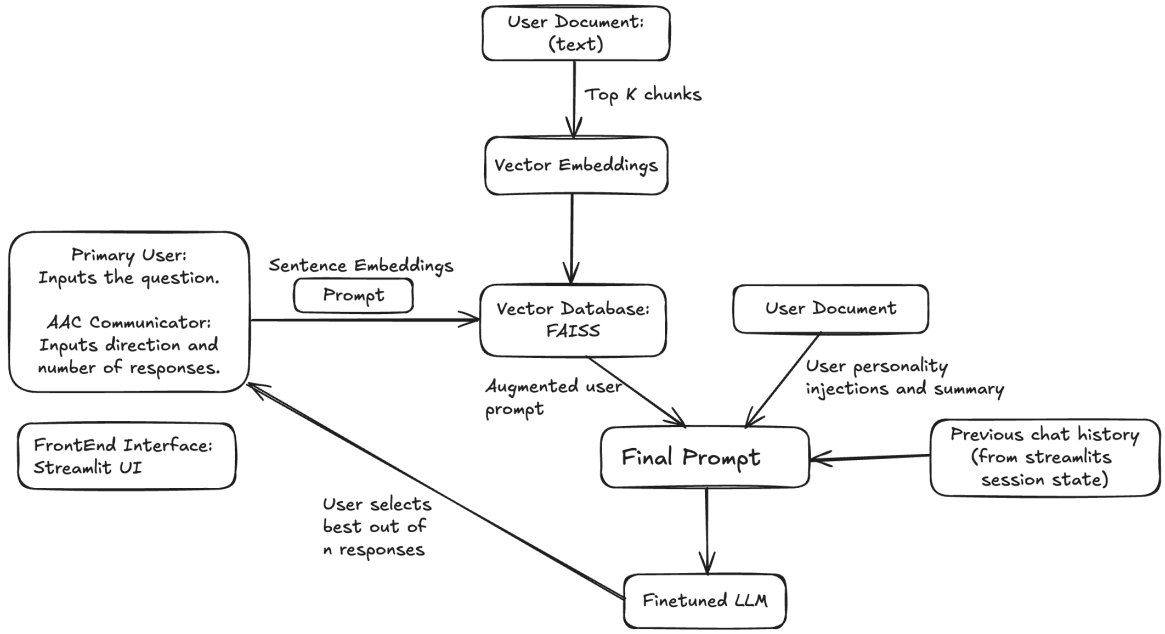

Figure 1: Solution Architecture

empathetical responses that are emotionally driven. For this purpose, we fine tune the LLM model on 2 particular datasets, the EmpatheticDialogues[1] dataset and the GoEmotions[2] dataset, both which offer a rich source of emotional and conversational data. A comparative analysis of both the datasets are provided in Table 1.

4 Solution Architecture

The workflow of the solution architecture can be seen in Figure 1.

The Llama model is finetuned on the datasets mentioned in section 3 for smoother and emotionally available conversations. We employed the unsloth library to aid us in the model setup and training. As the datasets were very large, we used only a chunk of the data to finetune the model. Lora was used to finetune the model as it was already pretrained on an instruction following dataset. This also ensured that the model didn't suffer from catastrophic forgetting.

We deployed a Streamlit UI for the front-end as it was an easy to use framework. The primary user inputs a question that they want to ask the AAC communicator. Then the AAC communicator can add direction to the responses by either choosing the preset buttons or adding their own input. The user can also choose the number of responses they want the LLM model to generate.

[1] https://paperswithcode.com/dataset/empatheticdialogues

[2] https://paperswithcode.com/dataset/goemotions

All of these inputs are then converted into sentence embeddings using all-MiniLM-L6-v2 model. The prompt entered by the primary user is used by FIASS to retrieve the most relevant chunks from the user data.

Finally the sentence embeddings from the user prompt, the chunks retrieved from RAG by FAISS, the users personality and summary, previous chat history, and AAC communicators direction are all concatenated together to form the final prompt. The final prompt with the number of responses is provided to the fine-tuned LLM model[3] to generate the responses. These responses are then provided back to the AAC communicator where they can choose the most favorable response in the front-end user interface.

4.1 RAG Document

The RAG document used for this Llama model is a JSON formatted document that consists of information about multiple users in a key-value pair format. Each key is a unique key made by concatenating the users name. Each key consists of the users name, communication style which is used to inject personality traits to the LLM, persona summary that encourages the LLM to add quirks about users to the responses and personal documents which contains relevant information about the users lives. The relevant information in the personal documents can contain an average day in their life, hobbies, past experiences, etc.

4.2 FAISS Vector Store

We used FAISS (Facebook AI Similarity Search) library as our vector store for user documents and retrieval due to its high performance, scalability and efficient similarity searching capabilities. The FAISS store vector embeddings of the RAG documents that are stored in a JSON file as chunks and enables rapid retrieval of most similar chunks for a given query. These chunks provide contextual grounding for the Llama model for generating accurate and coherent responses. For the baseline evaluations, we used a chunksize of 300 with a chunkoverlap of 50 and retrieved the top most relevant chunk for each input.

4.3 Prompt Engineering

The prompt engineering played a vital role in shaping the quality and relevance of the generated responses. we designed prompts that explicitly provided context retrieved from FAISS along with user intent enabling the responses to be highly personalized and have coherent outputs. The prompts provided to the LLM had a strict format that consisted of multiple elements.

The final prompt that was provided to the LLM model consisted of 5 different parts. The selected_user is a system prompt that provides the LLM with a user and their personality to perform personality injections. It also appends a small summary regarding the user to improve response personalization. The current_input_value consists of the question that is being asked to the AAC communicator. The current_influence_value consists of the direction in which the prompt must be generated. This is usually provided by the AAC communicator. The history_for_prompt consists of the chat history that is saved in the streamlit session that enables the user to have a continuous conversation. Finally the retrieved_docs consists of the chunks retrieved from RAG using FAISS. These 5 parts are concatenated together to provide an input to the LLM model.

4.4 LLM Model

When the user enters a prompt, it is converted into embeddings using the all-MiniLM-L6-v2 model which are used to retrieve the top relevant chunks from the RAG pipeline. A new prompt is constructed using the initial user provided prompt, direction provided by the AAC communicator, personality infections of the user, chat history if available and the chunks retrieved from the RAG pipeline. This newly constructed prompt is given as an input to the finetuned Llama model which then generates a response to give back to the user.

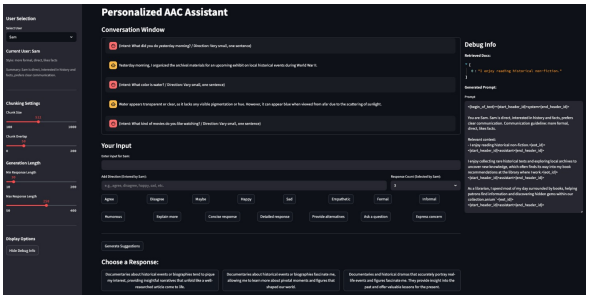

Figure 2: Application User Interface

[3] https://www.kaggle.com/datasets/mohankumarkakarla/finetuned/data

Query	Alex's Response	Sam's Response
User Personality	Slightly informal, friendly, uses humor	More formal, direct, likes facts
Summary	Alex is friendly, enjoys sci-fi (Star Trek), loves his dog Max and sister Chloe in London.	Sam is direct, interested in history and facts, prefers clear communication.
What are your hobbies?	I'm a total Trekkie at heart - when I'm not hanging out with my furry sidekick Max or catching up on the latest epics of life with my sis Chloe back in London!	I enjoy collecting rare historical texts and exploring local archives to uncover new knowledge, which often finds its way into my book recommendations at the library where I work.
Hi! What is your occupation?	Hey! By day, I moonlight as a part-time data analyst, but by night, I boldly go where no spreadsheet has gone before... just kidding, sorta!	As a librarian, I spend most of my day surrounded by books, helping patrons find information and discovering hidden gems within our collection.
What did you do yesterday morning?	Yesterday morning, I took Max for a galactic adventure... err, I mean, a walk around the block while sipping coffee and checking the news.	Yesterday morning, I organized the archival materials for an upcoming exhibit on local historical events during World War II.
What colour is water	Water's clear, like Captain Kirk's eyes when he's got a plan to save the galaxy! (Just kidding, it's transparent!)	Water appears transparent or clear, as it lacks any visible pigmentation or hue. However, it can appear blue when viewed from afar due to the scattering of sunlight.

Table 2: Responses generated for input queries for Alex and Sam. (Direction: "Very small, one sentence.")

5 Baseline Evaluations

The performance of the LLM model was evaluated using two types of evaluation metrics. The first type of evaluation metric is the automatic metrics in which we evaluated the LLM model using cosine similarity, BERTScore. The model achieved a cosine similarity of an average of 92% which shows very good performance and a BERT F1 Score of an average of 54% which shows room for improvement.

Cosine Similarity	BERTScore		
	Precision	Recall	F1
0.9202	0.4811	0.6034	0.5423

Table 3: Comparison of Cosine Similarity and BERTScore Metrics

The other type of evaluation metrics is the human evaluation in which we evaluated the generated responses on relevance, sincerity, understandability, fluency, and personalization. Based on human evaluations by 4 evaluators, the responses show high relevance to the input prompts provided. This is crucial for question answering and to prevent hallucinations. Sincerity makes sure that the responses feel genuine and authentic as opposed to generic, repeated responses. The model shows personalized responses for different users based on their personality traits. The responses are also relevant to the prompt from the user. The responses are very easy to understand and straight to the point but sometimes there might be unnecessary information added to the response. The responses are fluent for the most part. The model is able to generate responses in multiple emotions with personality traits showing high personalization which shows promising results in terms of personalized responses which are relevant to user documents and input prompts.

A few of the queries and responses are available in Table 2.

Conclusion

This project has explored the application of LLMs to create more personalized AAC systems, utilizing RAG and personalized prompting to generate dialogues that better reflect individual user char-

acteristics. The results of this work show the importance of continued research in this area, with future efforts focusing on ease of use and personalized solutions. By pursuing these avenues, we can move closer to developing AAC technologies that empower individuals with speech impairments to communicate with greater authenticity and autonomy.

References

https://paperswithcode.com/dataset/goemotions

https://paperswithcode.com/dataset/dailydialog

https://huggingface.co/docs/diffusers/en/training/lora

https://www.datacamp.com/blog/rag-advanced

https://blogs.nvidia.com/blog/what-is-retrieval-augmented-generation/

https://github.com/Tiiiger/bert_score