

Voice Based Spreadsheet Analyzer

Ayan Rajput¹, Ms. Meenu Garg²

Department of Information Technology, Maharaja Agrasen Institute of Technology, Delhi, India¹

Assistant Professor, Department of Information Technology, Maharaja Agrasen Institute of Technology, Delhi, India²

Abstract: In recent years, Artificial Intelligence has shown significant progress and its potential is growing rapidly. An application area of Artificial Intelligence is Natural Language Processing. Voice assistants can communicate with the users in natural language. Voice assistants are easy to use and can make many difficult tasks easy. But currently there is no voice assistant to analyze spreadsheet data and also it is hard to analyze spreadsheet data for people with non-analytics background.

All AI programs are reasoning programs. And, to the extent that they reason well about a problem, all possess some expertise at problem solving. Expert system has a knowledge base containing accumulated experience and an inference or rules engine which is essentially a set of rules for applying the knowledge base to each particular situation that is described to the program. The system's capabilities can be made better with additions to the knowledge base or to the set of rules.

In this project we are trying to make a voice-based spreadsheet analyzer which will be an expert system. This expert system will contain a knowledge base which will be used for query processing. It will take a spreadsheet as input then users will speak their queries, the queries will be converted into text and it will perform the necessary operation with the spreadsheet data, knowledge base and queries and will show the results accordingly.

INTRODUCTION

In this new age of technology and innovation, the use of artificial intelligence and machine learning has made our lives much easier. These technologies have been shown to benefit the public in various fields such as education, industry, online commerce, etc. and one of the most prominent is communication. Intelligent Virtual Assistant is a highly developed software with a powerful speech recognition system focused on audio signal processing in the system, translating it into text and performing the required task. Nowadays various intelligent virtual assistant is available for performing various tasks and automating different work.

An application that plays a major role in a person's life is an expert system. For example, there are Expert systems used to diagnose human disease, profit-related speculation, and logbook studies of motor vehicles individually. Some expert bodies are proposed to replace human authorities, while others are designed to assist them. Professional framework functions are part of a standard class of PC framework applications known as Artificial Intelligence.

Planning a pro-structure, man needs a data creator, a man who focuses on how human experts solve decisions and makes sense models in terms that the PC framework can get.

We are developing an intelligent virtual assistant which can analyze spreadsheet data using voice as input. It is not possible to design an intelligent system to solve this problem as it requires lots of sample data. An intelligent system learns from past data and solves the new problems based on that learning. Here we don't have any previous data available as it is a new type of technology so we are using expert systems which relies on predefined set of rules to solve a particular problem.

LITERATURE REVIEW

Speech recognition has a long history and several waves of great new things. Speech recognition, search, and voice commands have become quite common on computers, smartphones and portable devices.

A spreadsheet has always been a tool for making history. Decades ago, he introduced business software to personal computers, helping people save time on number-based activities, from budgeting to organizing Little League roster. Nearly 50 years later, the spreadsheet enters a new era, one in which we can replenish and renew using the Internet as a data source.

Nowadays there are various tools present for the automation of spreadsheet such as excel vba and google apps script but they require skilled professionals to correctly use them. There have been no studies around making a chatbot which even people having no or very less experience in data analytics can easily use to analyze spreadsheet while asking simple queries to get relevant insights from the data.

METHODOLOGY

To make a tool which can easily process spreadsheet data we require a programming language which is capable of handling and processing data very easily. Here python comes into play, python nowadays is heavily used for data analytics and visualization. Python also supports web application development through frameworks like Django, Flask,

etc. which are primarily being used to create backends APIs (Application Programming Interfaces). Here we are using python in our project for both of its applications. The frontend of the project is created using HTML, CSS and JavaScript and backend using Django (A Python Web Development Framework).

The frontend of the project is created using HTML, CSS, JavaScript and jQuery and XLSX libraries of JavaScript. jQuery is primarily a DOM (Document Object Model) manipulation library for the JavaScript which is also used for making and handling asynchronous AJAX (Asynchronous JavaScript and XML) requests to the server. JavaScript XLSX library is a library to read data from a excel spreadsheet converted into binary format and then convert it into JSON objects.

To get the voice input and its processing we use the Web Speech API. The Web Speech API enables you to add voice data into web apps. The Web Speech API has 2 parts: SpeechSynthesis (Text-to-Speech), and SpeechRecognition (Asynchronous Speech Recognition).

The frontend gets the excel, converts it into binary format and then converts it into JSON object and stores it on the frontend and displays the data in form of HTML tables to the user to left side of their screen. Then user presses the speak button and web speech API uses its text-to-speech engine to convert it into text. Then the text along with the excel data (JSON format) is sent to the backend for processing. If the backend runs successfully and responses with status code of 200, then we get the JSON object from the backend and use it to display the data to the user accordingly. If type of data is string, then it is shown to the user as it is, but if its type of the data is table then we create a HTML table using it and will show the table to the user on the right side of the screen. The user can again press the speak button again to generate a new input and the get the results.

The backend of the project is made using Django, which is a web framework for python, it handles the request to display the page to the client using Views and URLs. When the user makes a request to the /analyzer endpoint it returns the webpage to the user which is essentially the frontend of the project. When the user makes ajax call to server using the excel data (JSON format) and the voice input which is already converted into text it processes it using pandas and returns the results in JSON format.

The queries are divided into 3 categories:

- Sorting Queries
- Max/Min Queries
- Filtering Queries

Sorting Queries: If the query contains words like arrange, sort, etc. then it will be considered as a sorting query. First, we convert the queries and data columns into lowercase, then we iterate through each column to check if it is present in the query, if no match is found then we return unable to process the query. But if a column is matched, the data is sorted using that column using pandas, we also check if query contains the word descending, if it is found then sorting is done in reverse order. Then it sends response as type table as a JSON response.

Max/Min Queries: If the query contains words like maximum, minimum, greatest etc. then it is considered as a max/min query. First, we convert the queries and data columns into lowercase, then we iterate through each column to check if it is present in the query, if no match is found then we return unable to process the query. But if a column is matched we fetch the row containing max/min of the number and return it as type table as a JSON response.

Filtering Queries: If the query contains words like where etc. then it is considered as a filtering query. First, we convert the queries and data columns into lowercase, then only use the query on the right side of the filtering keyword (e.g., where) then we iterate through each column to check if it is present in the query, if no match is found then we return unable to process the query. But if a column is matched we look for words like greater than, less than, equal to etc. if these words are found then we take the data to the right side of the word and match it against the found column in each row of data. Then we return matched rows as type table as a JSON response.

RESULTS

The project was able to answer simple queries of the user of type Sorting Queries, Max/Min Queries and Filtering Queries. It is very quick to convert the excel spreadsheet into JSON format because it is processed on the frontend without any help of the server. The voice to text engine is working as expected. Suppose the table contains the column name, age, category, marks, year etc. then the project was able to answer queries like “give me the maximum of age”, “sort the data by name”, “arrange the data by marks descending”, “get the data where year is equal to 4” etc. It was unable to answer complex queries like “give me the top 3 students by marks”, “what is age of student with name Ayan”, and some queries where query is asked in a different format like “get student whose age is more than 23” which is not incorporated in the inference rules. It currently only supports a single sheet in a spreadsheet.

CONCLUSION AND FUTURE SCOPES

This is an attempt to make a voice-based spreadsheet analyzer to ease the work of the user in which user can give a spreadsheet as input and ask queries and it will answer those queries using the data from spreadsheet. It is able to answer

simple queries but unable to answer complex queries. It also fails when column names of spreadsheet are in a format which can't be spoken like column names which contains an underscore.

It has great potential to become a fully viable product if we increase the knowledgebase and add more rules in the processing of query. In the future we can add more types of queries, ability to process more than one sheet and combine multiple sheets to answer the queries and be able to return data in different formats rather than showing a HTML table.

REFERENCES

- Research on expert systems, *Jan 1982, Bruce G Buchanan, University of Pittsburgh*
- A Voice Based Assistant Using Google Dialogflow and Machine Learning *May 2021 International Journal of Scientific Research in Science and Technology DOI:10.32628/IJSRST218311*
- Neiffer, J. P. (2018). Intelligent personal assistants in the classroom: Impact on student engagement. *Ph.D. Dissertation. College of Education and Human Sciences, University of Montana. Nilsen, J. H*
- Røyneland, K. (2019). «It knows how to not understand us! » *A study on what the concept robustness entails in design of conversational agents for preschool children (Master's thesis).*
- Voice Assistants and Smart Speakers in Everyday Life and in Education George TERZOPOULOS, *Maya SATRATZEMI Department of Applied Informatics, University of Macedonia, Thessaloniki, Greece Email: gterzopoulos@uom.edu.gr, maya@uom.edu.gr*
- Automated Data Analysis Using Excel June 2007 DOI:10.13140/2.1.2381.2803 Edition: 1 Publisher: Chapman and Hall CRC Editor: Randi Cohen ISBN: 978-1584888857 Brian Bissett Department of the Treasury