Educational Administration: Theory and Practice
2024,30(4), 15117 - 15127
ISSN:2148-2403
https://kuey.net/
Research Article

Adaptive Web Accessing Tool For Visually Impaired People With Explainable AI

Vedant Kesharia[1*], Palak Shah[2], Kirtan Shah[3], Harshal Dalvi[4], Neha Katre[5] And Prachi Dalvi[6]

[1*]Dwarkadas J. Sanghvi College of Engineering, No. U, 15, Bhaktivedanta Swami Rd, opp. Cooper Hospital, Navpada, JVPD Scheme, Vile Parle, Mumbai, Maharashtra 400056, keshariavedant@gmail.com, https://orcid.org/0009-0005-9697-636X
[2]Dwarkadas J. Sanghvi College of Engineering, No. U, 15, Bhaktivedanta Swami Rd, opp. Cooper Hospital, Navpada, JVPD Scheme, Vile Parle, Mumbai, Maharashtra 400056, palak.aspire@gmail.com, https://orcid.org/0009-0004-0932-6995
[3]Dwarkadas J. Sanghvi College of Engineering, No. U, 15, Bhaktivedanta Swami Rd, opp. Cooper Hospital, Navpada, JVPD Scheme, Vile Parle, Mumbai, Maharashtra 400056, kirtan171003@gmail.com, https://orcid.org/0009-0008-0379-8501
[4]Dwarkadas J. Sanghvi College of Engineering, No. U, 15, Bhaktivedanta Swami Rd, opp. Cooper Hospital, Navpada, JVPD Scheme, Vile Parle, Mumbai, Maharashtra 400056, harshal.dalvi@djsce.ac.in, https://orcid.org/0000-0002-6995-5951
[5]Dwarkadas J. Sanghvi College of Engineering, No. U, 15, Bhaktivedanta Swami Rd, opp. Cooper Hospital, Navpada, JVPD Scheme, Vile Parle, Mumbai, Maharashtra 400056 neha.mendjoge@djsce.ac.in, https://orcid.org/0000-0001-8320-7071
[6]Fr. Conceicao Rodrigues College of Engineering, Fr. Agnel Ashram, Bandstand Promenade, Mount Mary, Bandra West, Mumbai, Maharashtra 400050 prachi.dalvi@fragne.ac.in, https://orcid.org/0000-0002-4278-868X

Citation: Vedant Kesharia, et.al (2024), Adaptive Web Accessing Tool For Visually Impaired People With Explainable AI , Educational *Administration: Theory and Practice*, *30*(4), 15117 – 15127
Doi: 10.53555/kuey.v30i5.8347

ARTICLE INFO

ABSTRACT

In today's digital era, websites serve as vital repositories of information, yet their predominantly visual format poses significant challenges for blind and visually impaired individuals. Existing web-accessing tools often demand a level of computer literacy that is not universally attainable among this demographic. To address these limitations, this study attempts to develop a novel web-accessing AI tool specifically tailored to the needs of blind and visually impaired users with varying levels of computer literacy. The envisioned web-accessing tool aims to empower blind and visually impaired individuals to access information from the web seamlessly using voice commands, thereby bypassing the barriers posed by visual interfaces. Unlike conventional tools, this solution prioritizes accessibility and ease of use, enabling users to access Wikipedia and other up-to-date information effortlessly. Using input speech recognition, techniques for accessing the latest information from various sources, and text-to-speech capabilities when it comes to audio output, the tool will revolutionize web accessibility for the visually impaired.

***Keywords*:** Explainable AI, Machine Learning, Multilingual, Visual impairment, Voice commands, Web accessibility

I. Introduction

However, for the millions of people across the world who are blind or visually impaired, there is a significant challenge in accessing this treasure trove of information[1]. According to the World Health Organization, about 253 million people globally have a visual impairment, and almost all face difficulty accessing digital content[2]. All over the world, there prevails a significant population, and such is the case in India, where the prevalence of blindness is significantly higher. Despite the rapid digitalization, a significantly large percentage of Indians with blindness face barriers to use the internet due to a lack of computer literacy. They face several challenges for accessing web content. More such challenges are involved with the accessibility needs in different languages. To address these challenges, this tool is designed to empower users by providing information in one place through seamless voice commands, effectively bypassing the barriers posed by visual interfaces. By leveraging advanced technologies such as speech recognition, the tool extracts information from websites and delivers it in a summarized, easily digestible format. Notably, an array of user-friendly voice commands enables effective control over the browsing experience by individuals. In this respect, for example, 'stop,' 'repeat,' 'ask the question,' 'help,' 'increase speed,' 'decrease speed,' 'Wikipedia,' etc. commands make it easy for website interaction while features like adjusting reading speed cater to individual preferences. More importantly, understanding the value of inclusivity, this tool integrates functionalities to help color-blind people, hence accessible to all. This research aims to improve web accessibility to ensure that blind and visually impaired people are not left behind in the digital age but rather included and empowered through digital interactivity. This MIR Labs, USA tool will address the peculiar needs of these groups, reducing the digital divide and ultimately ensuring equal access to information.

II. Literature Survey

The literature review to be discussed in this paper focuses on current web-accessing tools and assistive technologies developed for visually impaired individuals. Thus, the review is aimed at weighing up the strengths, weaknesses, and problems faced by the existing solutions for web accessibility that resulted from numerous research efforts and technological developments. The review lays the foundation for a new AI tool proposed for web access, specifically designed to meet the needs of visually impaired users by narrowing the existing accessibility gap to promote inclusivity in their online interactions. The work, "A Web Accessing Tool for Blind and Visually Impaired People Using Bahasa Indonesia"[3] uses speech recognition for input and text-to-speech for output in the primary information from websites employing transcoding techniques. However, it is essential to mention some limitations of this work: critical features like support of multiple languages are not implemented. Additionally, it recognizes that the Indonesian accent may pose challenges for comprehension among Indian users. The research paper titled "VOICE ASSISTANT – A REVIEW" [4] has the objective of discussing the development and usability of a Python-based Intelligent Software assistant designed to develop and evaluate a voice assistant application for performing various tasks, including checking weather updates, searching on Wikipedia, and making and receiving calls, particularly for blind, visually impaired individuals, and others, thereby enhancing their access to library resources. This tool gathers essential data from magazines, articles, books, and samples, and 15118ikipedi analogous applications. It follows the Scrum methodology for application development, integrating backlogs, sprints, and Scrum meetings. But the limitation in this paper is that it doesn't filter the unnecessary data while scraping 15118ikipedia. Hence, not the most optimal way to retrieve information from the web. The research paper titled "Voice Assistant for Visually Impaired People" [5] is about a system that has the functionality that provides a summary of website content and answers user questions based on this summary, utilizing a BERT model trained on the Stanford Question Answering Dataset. But this system lacks an integrated environment for blind users, making navigation challenging for them. Additionally, approximately 50% of the questions posed to the model remain unanswerable.

III. Methodology

A. System Design And Architecture

Figure 1 represents the system architecture of the Software, which adopts a modular client-server distributed design to ensure flexibility and scalability. At system startup, the main server initializes, serving as the primary entry point for user interaction. The system leverages REST APIs for seamless communication between client and server components, enabling effortless integration and remote access to the web modules. Users mainly communicate with the system using a speech to text interface, which makes use of JavaScripts SpeechSynthesis library to accurately transcribe their commands. Feedback and responses are delivered through text to speech conversion utilizing the library making sure that users, with impairments can easily access the information.[6] Core functionalities include processing user voice commands through a natural language processing (NLP) module [7], scraping web content based on the user's queries [8], and giving back responses based on the user's field of interest [9]. Moreover, the system interfaces with the Gemini model to service user queries, such as setting playback speed and offering access to help documentation for understanding commands. Besides the above feature set, the system provides flexibility for a module that is expected to provide user experience and personalization. When the request begins with, for example, "ask Wikipedia" or "ask the question," the system logs requests into an already specified file. This file is then passed to the in-house Machine Learning model. The model was trained on the user's field of interest, given the requests issued beforehand. It leverages past queries as training data and uses sophisticated algorithms to analyze and infer a user's preference and area of interest. In this manner, predictive ability endows the system with a modicum of intelligence for fine-tuning responses and recommendations based on the user's interests, consequently bettering the all-around user experience.

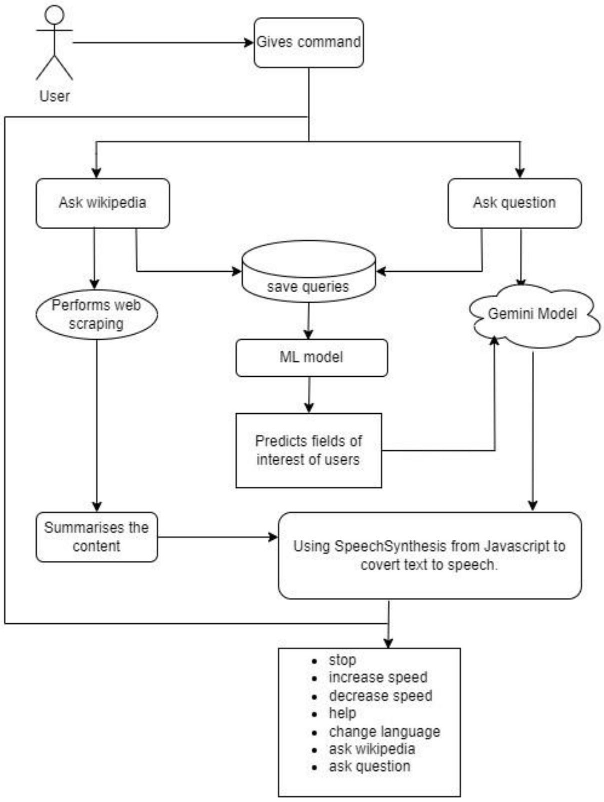

Figure 1. System Architecture Diagram

All modules are developed in Python and each module comprises customized scripts tailored to its specific features, such as question answering and summarization. The system's integration and cohesion are upheld through APIs developed using Flask, Figure 2 Flow diagram. Furthermore, the architecture is engineered to be operating system independent, ensuring seamless deployment and usage across different platforms for user convenience.

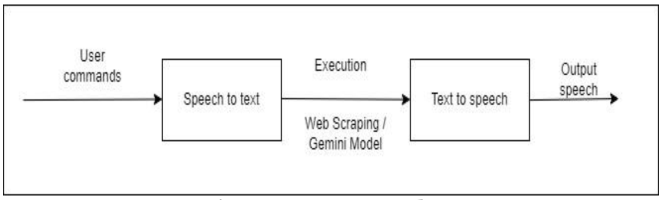

Figure 2. Data Flow

B. Implementation

The implementation of the application follows a user-centric approach, where the user interacts with the system primarily through voice commands, and the system responds by providing synthesized audio output. All the commands are listed in Table 1. The basic flow of the application is depicted in Figure 2. illustrating the seamless interaction between the user and the system.

Table 1. List Of Commands

Command	Function
wikipedia	Generates a summarized form of a Wikipedia query provided by the user
ask question	Allows the user to ask any question, with session data being saved for future reference.

change language	Changes the language in which the user can give input and receive output from the system.
increase speed	Increases the playback speed of the synthesized speech output.
decrease speed	Decreases the playback speed of the synthesized speech output
stop	Stops the execution of command.
help	Provides a list of available commands and their respective functions to assist users in navigating through the application.

1) Wikipedia Module:
Upon startup, the application prompts the user to give a command, expecting either "wikipedia" or "ask question" to initiate the desired functionality. If the user says "wikipedia," the system executes web scraping to retrieve relevant information based on the user's query. Subsequently, the system summarizes the retrieved content and speaks out the summary to the user using synthesized speech.

2) Question Module:
Alternatively, if the user opts for the "ask question" command, the system fires the Gemini model to obtain a concise answer to the user's query. The response from the Gemini model is then spoken out to the user, providing a quick and informative interaction experience.

3) Playback Speed Adjustment:
In addition to the core functionalities, the application includes support for adjusting playback speed to cater to user preferences. Users can increase or decrease the speed of synthesized speech output according to their convenience, enhancing the user experience.

4) Change Language Module:
The user activates the command by saying "change language." When this command is heard, the system prompts the user to choose the language by listing the given language. Ten languages are there for selection, and the first language in the list is English, as it is the default language and, hence, the most prominent choice. After English, Hindi follows; after Hindi, there is also a selection available for some other Indian languages. Once the user has selected a language from those on offer, the system switches the interface and functionalities to the chosen language. This change of language will be confirmed by speaking the selection back to the user using synthesized speech for the user to be alerted that the language update has been successful.

5) Accessibility Feature:
With inclusivity at its heart, the system has added three color modes that would specially cater to and help people with color-vision deficiencies. The application is intelligently designed to feature three different color modes, considering various types of color vision deficiencies. In a brilliant way, through the careful choice of certain hues of red and blue[10], these colors remain distinct from each other and are allowed to pass for common color vision deficiencies. However, beyond the primary colors, using secondary and accent colors is what makes for excellent user experience. The high-contrast combinations, like black text on a white background, will ensure readability even for colorblind users and will conform with other universal design principles. This application accommodates the specific needs of its diverse user base by prioritizing contrast and clarity in text presentation, making this an inclusive environment where all users can navigate and engage with ease. The implementation of the application puts user interaction and accessibility as top priorities by using voice commands and synthesized speech output to do just that, creating a seamless and natural-feeling user experience. The application is designed and incorporated with various functionalities that allow access to and retrieval of information very quickly, which also increases user convenience and accessibility.

C. Model Used
The machine learning(ML) model Figure 3 integrated into the application is the one big driver that enhances user experience and personalization. Using historical data about user queries, the machine learning model infers user interest and preferences by applying advanced algorithms in pattern analysis.

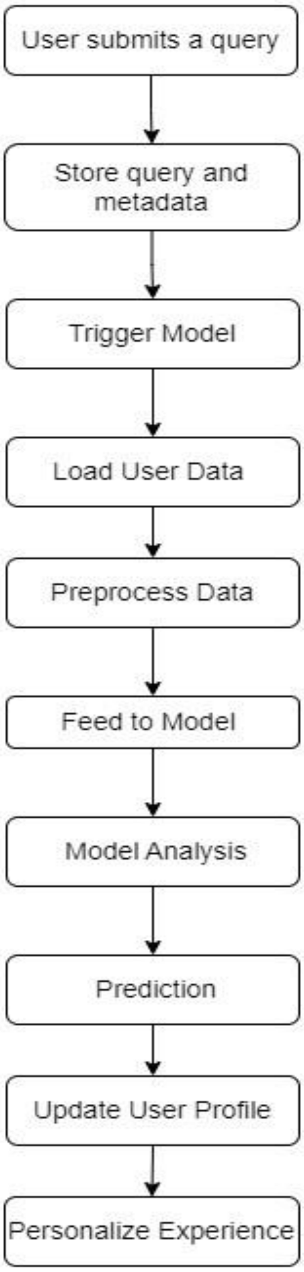

Figure 3. ML Model Flow Chart

The model would apply some mechanisms, such as natural language processing(NLP) and predictive analytics, to make predictions about the user's field of interest from their past interaction with the system. In the implementation, the application systematically logs all user queries into one file, thus ensuring a complete history of user activities. Each query is appended to this file in a structured format. This file is processed by the application from time to time or whenever a trigger is fired, and it passes the contents to the ML model for analysis. So, the file having those user queries is an input for the ML model. The model will then apply algorithms to the input so that it can be able to detect trends, patterns, and correlations in the data. With time, there is an iterative learning and improvement of the model as it becomes increasingly predictive. Now, this allows the model to keep generating an ever-more precise assessment of the user's field of interest. When the ML model is done analyzing, it outputs predictions or recommendations based on the user's query history.These are what the system uses to guide each response and interaction that is personalized to the user's interests and likings. This way, integration of the ML model is a watershed step in using user data to enhance the quality of interaction. Going much beyond simple pattern recognition, the model embodies a genuinely active and dynamic way of learning and adjusting to ensure that user experience keeps getting fine-tuned and optimized. It catches subtle differences in user behavior, making sense into actionable insights that drive much more meaningful interactions. It is the system's real strength that, besides relevance, it can continue to evolve with user preferences and understand future needs and patterns in advance anticipating behavior and interest in understanding and predicting the future places this application at the very edge of personalized digital experiences. It sets a new standard in the engagement and satisfaction of the user in today's increasingly data-driven world, seeing that users receive what is relevant to their current interests and needs. This provides an app that meets and, through time, further enhances satisfaction, which is why it's a crucial tool for delivering superior, user-centric service and engagement. In essence, having the ML model in the application goes a long way in making the application proactively personalized and responsive, improving user experience. By systematically tracking user requests and using data in combination with advanced analytics, the application can provide very relevant and targeted content, thus enhancing user engagement and satisfaction.

1) Feature Importance Analysis of the Logistic Regression Model

The following section deals with the importance of logistic regression used for the prediction of user interest. The model is applied to the vectorized user query using TF-IDF, and the relative importance of the term is the coefficient given by the logistic regression model. From Figure 4, a horizontal bar graph is used to visualize feature importance. The horizontal axis represents the feature's importance score, with higher values indicating greater influence on the model's predictions. The vertical axis lists features (words/terms) in descending order of importance.

(a) Observations:

(1) Common Words: Interestingly, common words such as 'the,' 'in,' 'is,' and 'of' have high importance scores. While high occurrence would indicate they would pull many predictions in their direction, they could also be stopwords thrown out in the model refinement.
(2) Domain-Specific Terms: Importantly, domain-specific words including ‘invest’, ‘rate’, ‘train’, and ‘work’ showed high importance and could be vital for classifying user interest in domains.
(3) Low-Importance Features: At the lower end of the graph, features with low importance scores appear. These are probably words with relatively low-frequency counts and do not play a substantial part in the model's prediction, such as 'telecommunications,' 'veganism,' and 'education.'

(b) Key Findings:

The feature importance visualization offers valuable insights into the model's inner workings:
(1) Stopwords: From the high importance of common words, it seems there would be some benefit to stopword removal in pre-processing. This would enable the model to focus on other, more meaningful terms.
(2) Feature Engineering: Further exploration of alternative techniques for feature engineering might benefit in a way that the model can distinguish very different user query types.
(3) Regularization: Perhaps regularization techniques can help prevent overfitting, more so when common words overpower the decision-making process in the model.

2) Explainable AI (XAI) for Logistic Regression Model Interpretation

This section explains how to use Explainable AI when applying logistic regression models to understand user interests. By utilizing the feature importance analysis, insights are then obtained into the working mechanisms of the model and increase transparency.

(a) Feature Importance Visualization as an XAI Technique:

In Figure 4, the research applies a horizontal bar graph to represent the feature importance of the logistic regression model. This visualization is an XAI technique in the following manner:
(1) Enhancing Transparency: The graph reveals the model's decision-making process. It further exposes features (words/terms) in the user queries that are TF-IDF vectorized, which majorly hold importance in the model predictions. This kind of transparency makes stakeholders interpret the outputs from the model.
(2) Explaining Feature Contribution: The graph helps with the relative contribution of words or terms in the dataset because of the importance scores given to each feature. It lets one know the features that are most influential and the reasons for specific predictions by the model.
(3) Aiding Model Interpretation: It provides users with visual explanations that facilitate the trust and validation of the model's interpretation. For instance, when the model is classifying user queries into different fields, an idea of which terms are more informative will help a lot in understanding user interests.
(4) Facilitating Debugging and Improvement: Sometimes, a graph of feature importance can be used for debugging purposes. If common stop words like 'the,' 'in,' and 'is' seem to be very important, then this may be a pointer to further data preprocessing steps, like removing stop words, to enhance model performance.

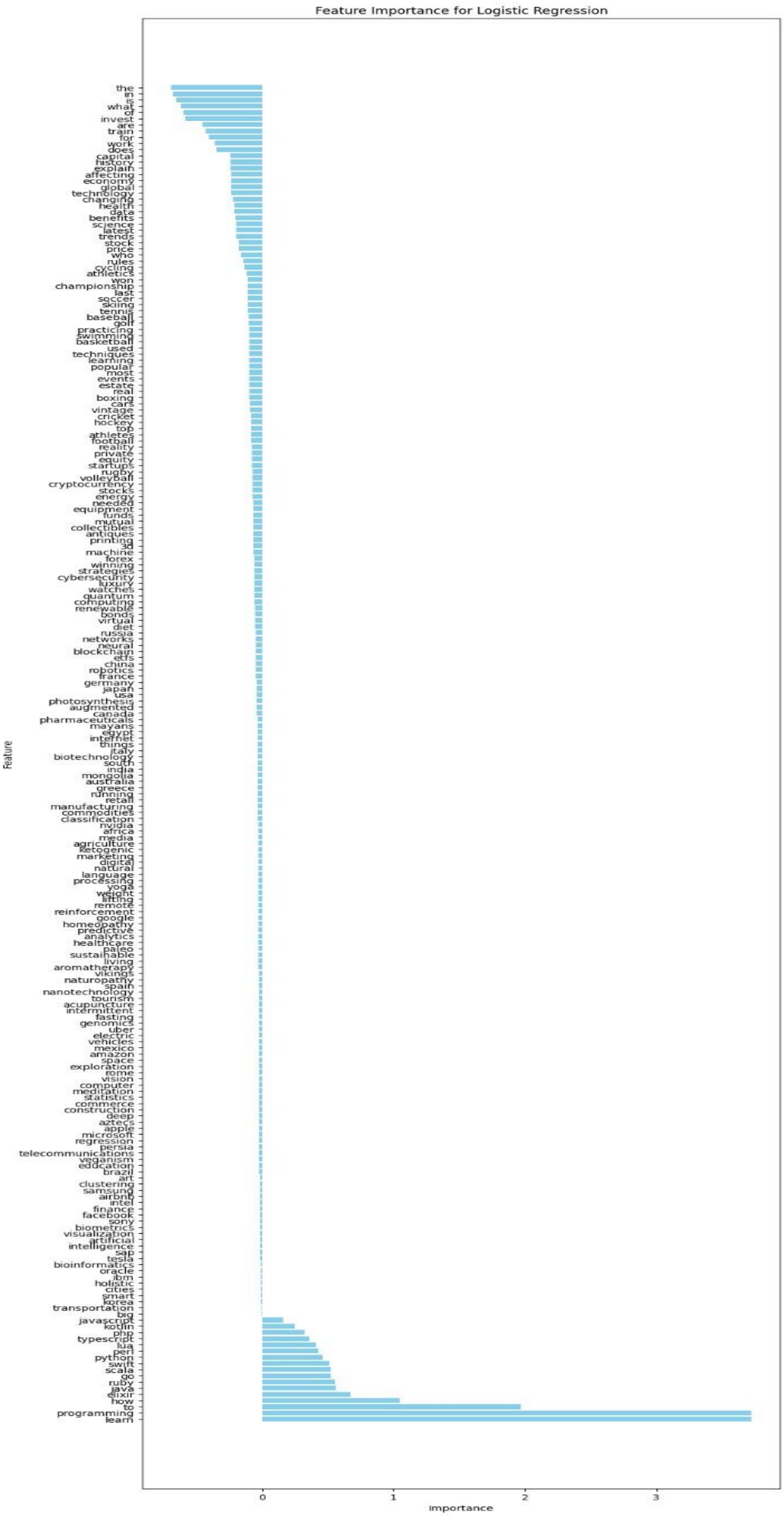

Figure 4. Feature Importance Graph

3) SHAP - Summary Plot for Explainable AI

This part explores the use of SHAP (SHapley Additive exPlanations) as one of the techniques in Explainable AI for interpreting a logistic regression model applied during user interest prediction. The SHAP summary plot in Figure 5 gives us information on the importance of features, interaction effects, and overall contributions to the model's predictions.

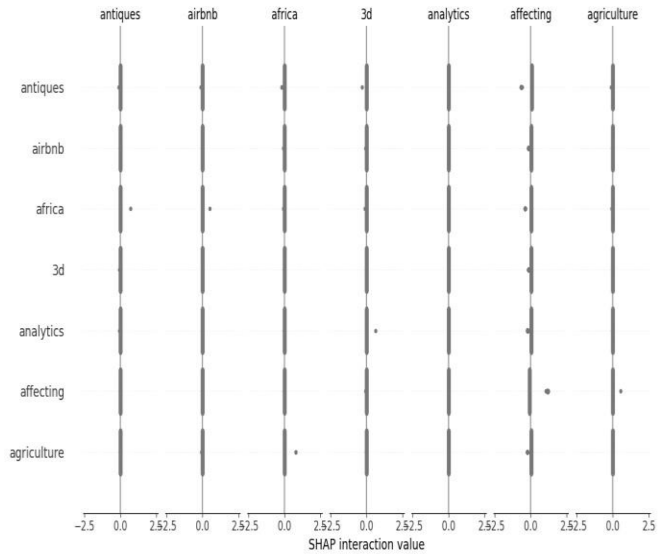

Figure 5. SHAP for XAI

The SHAP summary plot gives an essential insight into what the logistic regression model performs for user interest prediction. This plot provides a visual grasp of how the different features—words or terms— of the vectorized user queries in TF-IDF contribute to the model's predictions. The y-axis, in a plot, represents the features, and the x-axis represents interaction values from SHAP. These SHAP values measure how the effect of each feature is moving a given prediction from the average prediction given the features of a user query. Positive SHAP implies that an increase in the feature value being discussed would move the prediction away from the average. Negative SHAP means a shift toward the average. Each dot in a SHAP summary plot represents a SHAP value of one feature for one user query (instance) from the test set. The position on the x-axis of the dot reflects the magnitude and direction of this SHAP value in determining whether the given feature has a positive or negative influence on the prediction for that query. How consistently an important feature influences the model output across different instances is given by the spread of the dots along the x-axis. The SHAP summary plot provides insight not only into the importance of the individual features but also into the nature of feature interactions. Features that have larger SHAP values bring more influence to the prediction of a specific instance. Features with more spread dots should imply more interaction with other features. Such a feature interaction could be complex, which in turn would make the effect of the feature differ with different user queries. Several key points can be realized from this SHAP summary plot. I've selected features that likely will have a significant effect on predictions: 'antiques,' 'Airbnb,' 'Africa,' and 'analytics.' and how the SHAP values for those features are distributed. For example, the feature 'antiques' might have an extensive range of SHAP values, from -2.5 to 2.5. That is to say, the influence of 'antiques' on a particular prediction may be so powerful and even further differ from another query to push the prediction either in a good direction or a bad one. The SHAP summary plot has many advantages for XAI development. Visualizing feature importance and interaction effects makes it understandable which features are influential and how they affect model predictions. For instance, this kind of transparency assures how the model arrived at a decision. Based on the SHAP values' analysis, features that have more influence on the user interest prediction can be found and, in turn, can be helpful to information for feature engineering and model improvement. Lastly, from the SHAP summary plot, it can be observed how the features interact; hence, it gives greater insight into the hidden, complicated process of decision-making in the model.

IV. Results and Discussion

In this segment, the outcome of the performance evaluation of the web-accessing tool is described based on the responsiveness and processing of the voice commands given in the English and Hindi languages. The section has been categorized according to the type of command and language, with applicable averages and standard deviations. The results below are presented in seconds, where the time outside the brackets indicates the cumulative time, and the time inside the brackets represents the time taken between different steps.

A. English Language

1) Result 1:

(a) Ask command recognition: 1.8s
(b) Question recognition: 3.3 (1.5)s
(c) Answered: 5.1 (1.8)s

2) Result 2:
(a) *Wikipedia command recognition:* 1.6s
(b) *Query recognition:* 3 (1.4)s
(c) *Answered:* 4.9 (1.9)s

3) Result 3:
(a)Ask command recognition: 2.0s
(b)Question recognition: 3.55 (1.55)s
(c) Answered: 5.1 (1.55)s

4) Result 4:
(a) *Wikipedia command recognition:* 1.7s
(b) *Query recognition:* 3.94 (2.24)s
(c) *Answered:* 5.01 (1.07)s

B. Hindi Language
1) Result 5:
(a)Ask command recognition: 1.8s
(b)Query recognition: 3.68 (1.88)s
(c) Answered: 5.71 (2.03)s

2) Result 6:
(a)Ask command recognition: 1.72s
(b)Query recognition: 3.4 (1.68)s
(c) Answered: 5.45 (2.05)s

3) Result 7:
(a) *Wikipedia command recognition:* 1.6s
(b) *Query recognition:* 2.51 (0.91)s
(c) *Answered:* 5.51 (3)s

These results were generated through performance testing of the tool from initiation to recognition and response of a voice command. The performance results have demonstrated that, for varied types of queries and languages, the tool performs uniformly, hence enhancing web accessibility by users with visual impairments. Moreover, extremely low latency was experienced during the tests, ensuring the tool presented responses to voice instructions well. Logistic regression is employed to analyze the data and predict the user's field of interest based on their past interactions with the system. The following table summarizes the performance metrics of the model.

Table 2. Performance metrics

Accuracy	Weighted average Precision	Weighted average Recall	Weighted average F1 Score
97%	98%	97%	97%

C. Explainable AI for User Interest Prediction
LIME (Local Interpretable Model-Agnostic Explanations) is employed as an XAI technique to interpret the predictions made by the logistic regression model for user interest prediction. LIME facilitates explanation by creating a local surrogate model that approximates the original model's predictions in the vicinity of a specific query.

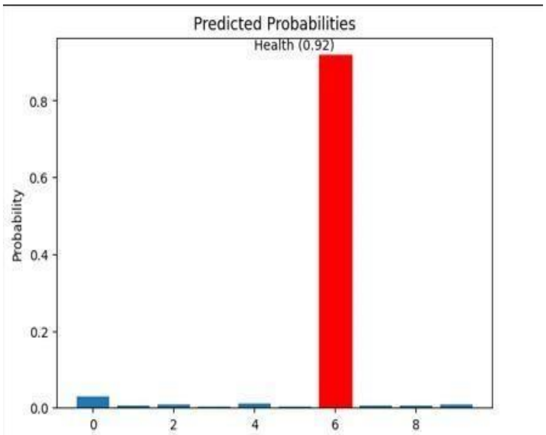

Figure 6. Class Probability Bar Chart visualized with LIME

1) LIME for Query Explanation

Consider the query, "What are the health benefits of yoga?". The logistic regression model predicts the class "Health" with a probability of 0.92 as shown in Figure 6. LIME works in three stages to explain this prediction:

(a) *Prediction by Logistic Regression Model:* The initial step involves the logistic regression model generating a prediction for the query.

(b) *Local Perturbation and Explanation Model Fitting*: LIME creates perturbed samples around the original query and observes the corresponding predictions from the model. LIME then fits a simple interpretable model (e.g., a linear model) to these perturbed samples to understand the features most influential for the original prediction.

(c) *Interpretation of the LIME Explanation:*

(1) Class Probabilities: The graph visualizes the probabilities assigned by the model to different classes for the query. In this case, the red bar indicates the highest probability class ("Health" at 92%). LIME sheds light on why this class received the highest probability by revealing how strongly the model associates the query with the "Health" category based on the features present in the query.

(2) Feature Importance: While the graph itself displays the predicted probabilities, LIME typically provides additional insights into which features (words or phrases) in the query significantly impaocted the prediction. For instance, for the query "What are the health benefits of yoga?", words like "health" and "benefits" are likely to be the key features contributing to the high probability assigned to the "Health" class.

(3) Local Fidelity: LIME ensures the explanation aligns with the model's behavior for the specific query, providing confidence that the prediction is based on relevant features.

V. Conclusion

The need for this research paper in today's digital world is met by creating a new web accessing artificial intelligence (AI) tool that specially targets the blind and visually impaired people with various levels of computer illiteracy. This paper highlights the main challenges that arise when this category of persons attempts to access digital content as well as stresses on the need to create accessible and userfriendly alternatives to fill existing gaps. Advanced technologies such as voice recognition, web scraping and text-to-speech conversion are harnessed to facilitate accessing content from the web through voice commands thus bypassing visual interfaces that hinder access. The tool will be designed with accessibility and usability in mind enabling blind and low vision users to easily access information from the web even in languages other than English. The research paper provides a comprehensive review of existing web-accessing tools and assistive technologies tailored for visually impaired users, highlighting their strengths, limitations, and areas for improvement. Drawing upon insights from the literature survey, the paper proposes a modular client-server distributed design for the tool's system architecture, ensuring flexibility, scalability, and seamless integration of diverse functionalities. In order to provide an organic user experience, this tool utilizes voice interaction and synthesized speech output. The system is smoothly linked with features, like web scraping, answering questions, language switching, storing user search history and adjusting playback speed to meet user requirements and preferences. In the paper, a machine learning (ML) model is introduced that is integrated into the application to enhance user experience and personalization. By analyzing user query data the ML model can make predictions about users interests and facilitate proactive content suggestions and tailored interactions. In general, this research paper enhances web accessibility for blind people or those who have low vision hence promoting further inclusion in their digital interactions. By targeting this group's special requirements while leveraging refined technologies; the said web-accessing tool has potential to transform web accessibility thus ensuring equal information access among all.

References

[1] World Health Organization. (2023). Blindness and vision impairment. Retrieved from https://www.who.int/newsroom/factsheets/detail/blindn ess-and-visual-impairment

[2] World Health Organization. (2012). Control and prevention of blindness and deafness. Retrieved from https://www.emro.who.int/control-and-preventions-ofbl indness-and-deafness/announcements/globalestimateso n-visual-impairment.html

[3] F. Kusumaningayu and M. Ayu. “A web accessing tool for blind and visually impaired people using Bahasa Indonesia,” in *Proceedings of the Second International Conference on Informatics and Computing (ICIC)*, pp. 1-6, Nov. 2017

[4] A. A. Patil, R. A. Gavali, and S. S. Shetty. “Voice Assistant - A Review,” *International Journal of Engineering Applied Sciences and Technology*, vol. 5, no. 11, pp. 157-158, March 2021.

[5] I. Borkar, A. Shaikh, S. Jadhav, V. Khandade, and B. Nagpure. “Virtual Assistant for the Blind,” *International Journal for Research in Applied Science and Engineering Technology (IJRASET),* vol. 10, no. 12, pp. 2049-2053, Dec. 2022.

[6] R. Sangpal, T. Gawand, S. Vaykar, and N. Madhavi. “JARVIS: An interpretation of AIML with integration of gTTS and Python,” in *Proceedings of the 2019 2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*, pp. 486-489, Jul. 2019.

[7] A. Bhalerao, S. Bhilare, A. Bondade, M. Shingade, and A. Deshmukh. “Smart Voice Assistant: A universal voice control solution for non-visual access to the Android operating system,” *International Research Journal of Engineering and Technology (IRJET)*, vol. 4, no. 1, Jan. 2017.

[8] Naik, K. (2019, December 5). Perform web scraping on Wikipedia - Data Science [Video]. YouTube. https://www.youtube.com/watch?v=NeAhhBRHy4E

[9] Z. Gharibshah, X. Zhu, A. Hainline, and M. Conway. “Deep Learning for User Interest and Response Prediction in Online Display Advertising,” *Data Science and Engineering*, vol. 5, pp. 12-26, 2020.

[10] Kumar, K., & Uttekar, P. S. (2024). What colors do you see if you’re colorblind? MedicineNet. Retrieved May 16, 2024, from https://www.medicinenet.com/what_colors_do_you_se e_if_youre_colorblind/article.htm

[11] H. Nguyen, H. Cao, V. Nguyen, and D. Pham. “Evaluation of Explainable Artificial Intelligence: SHAP, LIME, and CAM,” *ResearchGate, FAIC 2021*, May 2021.

[12] S. R. Islam, W. Eberle, S. K. Ghafoor, and M. Ahmed. “Explainable Artificial Intelligence Approaches: A Survey,” arXiv, 2021.

[13] S. Mohseni, N. Zarei, and E. D. Ragan. “A Multidisciplinary Survey and Framework for Design and Evaluation of Explainable AI Systems,” *ACM Transactions on Interactive Intelligent Systems (TiiS)*, vol. 11, no. 3-4, pp. 1-45, Sep. 2021.

[14] P. Gohel, P. Singh, and M. Mohanty. “Explainable AI: Current Status and Future Directions,” arXiv, 2021.

[15] A. Holzinger, A. Saranti, C. Molnar, P. Biecek, and W. Samek. “Explainable AI Methods - A Brief Overview,” in xxAI - Beyond Explainable AI, A. Holzinger, R. Goebel, R. Fong, T. Moon, K. R. Müller, and W. Samek, Eds., *Lecture Notes in Computer Science*, vol. 13200, Springer, Cham, 2022.

[16] V. Vishwarupe, P. M. Joshi, N. Mathias, S. Maheshwari, S. Mhaisalkar, and V. Pawar. “Explainable AI and Interpretable Machine Learning: A Case Study in Perspective,” *Procedia Computer Science*, vol. 204, pp. 869-876, 2022.

[17] Enderle, R. (2023). *Generative AI could be a critical tool for the visually impaired*. Computerworld. Retrieved from https://www.computerworld.com/article/1632153/gener ative-ai-could-be-a-critical-tool-for-the-visually-impair ed.html

[18] J. Wang, S. Wang, and Y. Zhang. “Artificial Intelligence for Visually Impaired,” *Displays*, vol. 77, Art. no. 102391, 2023.

[19] Alexiou, G. (2023). *How artificial intelligence transformed technology for the blind in 2023*. Forbes. Retrieved from https://www.forbes.com/sites/gusalexiou/2023/12/15/ho w-ais-explosive-growth-opened-up-the-world-for-the-bl ind-in-2023/

[20] O. Bendel. “How Can Generative AI Enhance the Well-being of Blind?,” arXiv, 2024.