An Improvised GPS Navigational Application to Reduce Abstractness by using Artificial Intelligence

Mohammad Omar Faheem*, Neeshant Badrinath Prakash, Luqman Kamar Sham, and Narendra Kumar Aridas

Department of Electrical Engineering, Faculty of Engineering, Universiti Malaya
50603 Kuala Lumpur, Malaysia

Corresponding email: omarfaheem0502@gmail.com

Abstract. This article addresses the confusion caused by existing GPS systems, which often rely on complex schematic arrows, leading to missed junctions and heightened accident risks. A survey of 150 drivers revealed that 62% lost time due to navigation issues, 80% frequently missed junctions, and over 70% needed additional assistance. Furthermore, the literature suggested that current GPS systems place a significant cognitive burden on drivers, impairing their ability to make timely decisions. These findings highlighted the need for a more intuitive navigation system. To tackle this issue, the article proposes an AI-based voice navigation system, using speech recognition, natural language processing, and Large Language Models (LLMs) to minimize cognitive load and enhance driver safety. Testing against specific metrics derived from driver feedback suggests that the solution can reduce confusion and abstractness of the GPS systems to a certain extent. However, road tests have not been conducted. The project's outcome underscores the importance of improved navigation aids to reduce driver distraction, enhance road safety, and prevent missed junctions, ultimately improving the overall driving experience.

Keywords. GPS; Navigation systems; NLP; Speech recognition; Fine-tuning LLMs.

1. Introduction

Interchanges are vital components of freeway and highway systems, facilitating access to nearby urban, suburban, and rural areas. These interchanges help drive land development nearby, resulting in the emergence of commercial spots such as hotels, gas stations, and fast-food restaurants [1]. To give a quick review, exit ramps are classified as lanes on the highway/interchange that diverge drivers to another part of the highway or the interchange itself. In contrast, side streets or access driveways are at the neighborhood of these interchanges that connect to local businesses, small neighborhoods, other business properties, etc. These are in the immediate vicinity of interchanges and may lead drivers to enter exit ramps and drive in the wrong direction. Wrong-way driving (WWD) occurs when vehicles travel in the opposite direction of the intended traffic flow on divided roadways, including interstates, freeways, expressways, and their access points [2]. A review of crash data from the Fatality Analysis Reporting System (FARS) between 2004 and 2011 showed that, on average, 359 lives were lost annually in 269 fatal WWD incidents [3]. Drivers, especially non-local, often use GPS navigation systems to help around with direction and navigations on road. Moreover, a previous study showed that in addition to human errors, WWD is highly susceptible to certain interchange types (eg. Partial cloverleaf) [4]. Figure 1 depicts what WWD looks like on the road.

With the advent of more reliable technology in navigation, drivers usually use popular GPS systems for assistance in directions, particularly when entering or exiting freeways. These GPS systems are often equipped with audio functionality that supplements the navigation experience by providing ‘timely’ guidance before a turn. However, a study [5] in 2016 compared the then-most popular GPS apps in their effectiveness in providing timely instructions on interchanges, focusing exclusively on exit ramps as previous studies have suggested them to be the main entry points for WWD. The study concluded that none of the GPS devices in consideration guided the drivers accurately when the distance between the access point and the exit ramp was less than 100 feet. Additionally, the likelihood of taking the wrong direction is very high when the distance between an exit ramp and a nearby access point is less than 350 feet if the driver is relying on GPS voice instructions [5]. These conclusions imply a need for a more timely and enhanced guidance navigation system.

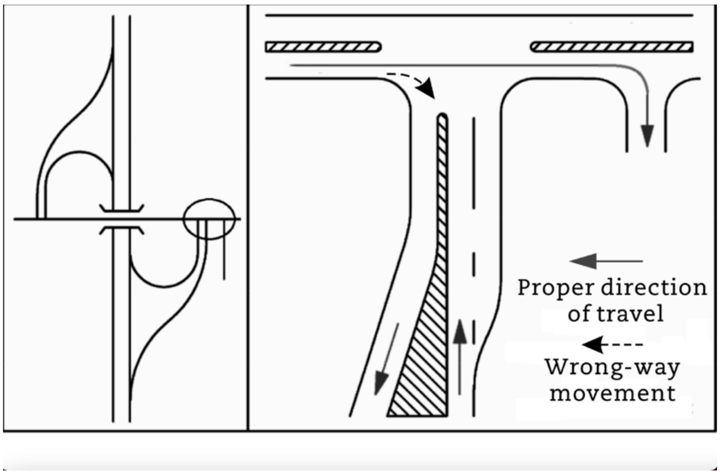


Figure 1. Representation of WWD [24].

This paper proposes a navigation app that tackles the need as mentioned earlier. Although the previous study solely focused on WWD events, a subjective test involving a questionnaire is conducted to confirm if the same problems persist in current navigation systems as expressed in [5] and to extract customer needs. These needs are cross-verified with a study that asserts which factors determine user acceptance of mobile navigation apps.

Based on this, an app is proposed with a set of functionalities that leverages the current developments made in the field of Artificial Intelligence, and its prototype development is outlined. Finally, the app is tested for its latency and its metrics are compared with previous studies. Additionally, the acceptance of the app by the public is considered using insights from another study.

Recent innovations in interactive audio navigation (IAN) systems have greatly improved support for differently-abled users, particularly the visually impaired. Integrating technologies like ultrasonic sensors, OCR, and AI-driven sound feedback, IAN systems now offer enhanced real-time object location and text identification. A study [5] highlights these advancements, especially for visually impaired users, utilizing ultrasonic and infrared sensors along with text-to-speech synthesis for effective navigation both indoors and outdoors. Another study [6] examines the integration of visual and auditory inputs for better environmental understanding, though it faces challenges in synchronizing data in real-time. Additionally, research on micro-drones for traffic data collection [7] introduces a novel method to optimize routes, despite concerns over data accuracy and privacy. This approach, though focused on traffic data, offers valuable insights for developing multi-input navigation technologies.

These days, mobile apps like Google Maps, Apple Maps, and Waze are widely used for navigation, offering both audio instructions and visual map guidance. Waze provides detailed routes and customization, Apple Maps excels in finding and learning about local businesses for iPhone users, and Google Maps is versatile across various modes of transport. This paper proposes an app that builds on these functionalities while addressing previously mentioned issues.

2. Methodology

2.1. Concept Development

Since Jalayer's study [5] is somewhat outdated, a recent survey was conducted to assess if navigation issues persist with modern apps like Google Maps, Waze, and Apple Maps (Survey results can be seen in Appendix 1.3). A total of 150 drivers participated. The results showed that 48% reported frequently missing turns and junctions despite confidence in their driving skills. Additionally, 44% of regular users found the apps reliable only "sometimes." It could be argued what 'sometimes' means here since it's subjective from person to person. However, the point to be highlighted is that regular users of these apps found them not helpful many times in their subjective connotation. These findings confirm that the problem still exists within drivers.

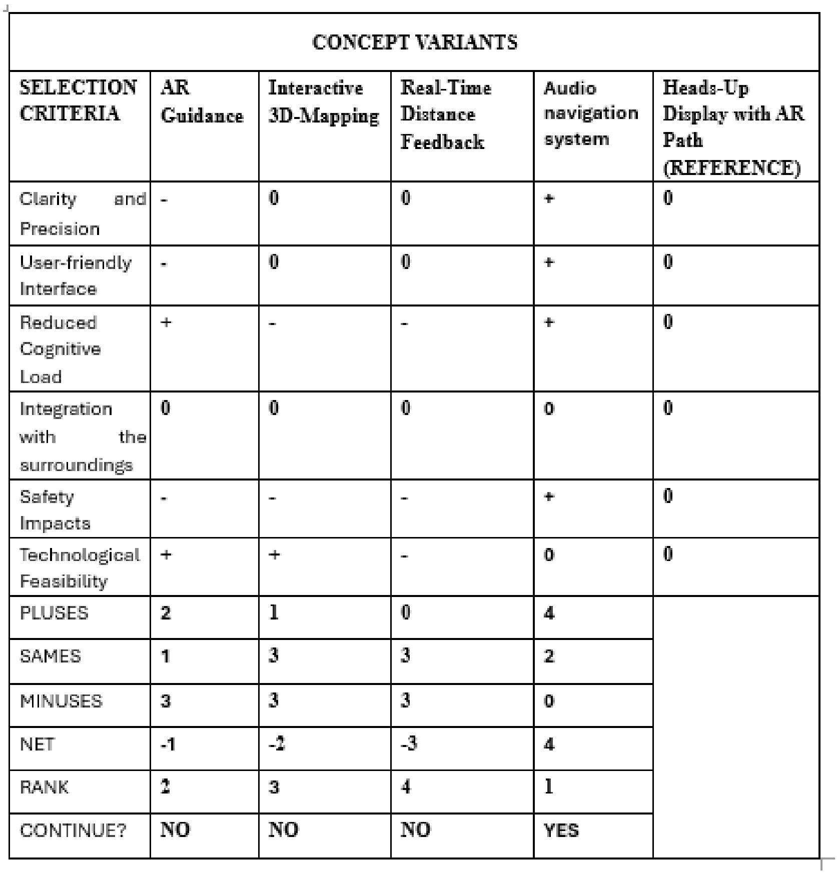
To exactly understand what problems drivers, face whilst navigating with these apps, another subjective questionnaire-based survey was conducted to extract what the drivers are looking for. Around 45 drivers took part in it. This questionnaire sought to explore the reasons why drivers took wrong turns, what sort of confusion developed at critical intersections on the road whilst using these apps and asked them to provide an example from their experience. Judging from the needs of the participating drivers, it was plausible to divide their needs into 3 main problems:

- Lack of proper lane guidance during turns/diversions/highway exits: Many drivers reported that for road diversions or for upcoming exits, they were not sure which lane to stay on and hence missed the turn because it was impossible to make the turn at the last minute.
- Instruction too abstract to understand while driving: Some drivers reported that information like "after 300m, take right" was too abstract to understand. This makes intuitive sense because how is a driver going to judge how much 300m is when driving!
- Lack of proper traffic updates: Some drivers also mentioned that due to the lack of traffic updates, they had to take routes that caused them to lose good time.

2.2 Concept Generation and Selection

Based on the needs identified from the survey, there is a need for a system that could potentially enhance driving safety and address the key challenges drivers face with existing navigation apps. One idea is to provide clear spoken directions, reducing the need for drivers to look at a screen, with a focus on a pleasant, easy-to-understand voice. To improve clarity, the system could use context-aware guidance, incorporating nearby landmarks for better real-time decisions, and lane-level precision to reduce missed turns and last-minute lane changes. Real-time traffic updates could enable dynamic route adjustments, ensuring smoother journeys. The design phase is also considering a user-friendly interface with voice commands, customization options, and quick response times for improved travel efficiency. Potential solutions based on brainstorming include various proposed technologies and methods. The following possible solutions are proposed:

- The Audio Navigation System **-** An audio navigation system has several algorithms that typically involves several key steps, including data collection, preprocessing, feature extraction, model training, and inference.
- AR Guidance - Add an augmented reality (AR) overlay on the windscreen so that distance indicators are shown right there on the road. For instance, the system may project a marker on the road that indicates the precise location of the turn, rather than stating, "Turn right in 200 metres.
- Interactive 3D Mapping **-** Gives users access to an interactive 3D map so they can virtually explore the route before setting out on it. Users can become more acquainted with impending curves and intersections by doing this.
- Real-Time Distance Feedback **-** As the vehicle gets closer to a turn or destination, provide continuous, real-time distance input. This might entail a graphic representation that dynamically changes the remaining distance and time to the next maneuver on the dashboard or Head-Up Display (HUD).



CONCEPT VARIANTS					
SELECTION CRITERIA	**AR Guidance**	**Interactive 3D-Mapping**	**Real-Time Distance Feedback**	**Audio navigation system**	**Heads-Up Display with AR Path (REFERENCE)**
Clarity and Precision	-	0	0	+	0
User-friendly Interface	-	0	0	+	0
Reduced Cognitive Load	+	-	-	+	0
Integration with the surroundings	0	0	0	0	0
Safety Impacts	-	-	-	+	0
Technological Feasibility	+	+	-	0	0
PLUSES	2	1	0	4	
SAMES	1	3	3	2	
MINUSES	3	3	3	0	
NET	-1	-2	-3	4	
RANK	2	3	4	1	
CONTINUE?	NO	NO	NO	YES	

Figure 2. Need-Constraint Matrix

Keeping these solutions in mind, the following constraints exist in the development of any of these prototypes:

- **Environmental and Sustainability Constraints:** Prioritize eco-friendly practices throughout the product's lifecycle by using recyclable or biodegradable materials, reducing energy consumption, and incorporating energy-efficient algorithms. These measures contribute to sustainability and environmental conservation.

- **Economic Constraints:** The design must focus on minimizing material costs and optimizing production efficiency to ensure the navigation system remains affordable. Smart design choices like component simplification and streamlined assembly help keep costs low, allowing for competitive pricing while ensuring sustainable revenue for ongoing development and maintenance.
- **Health and Safety Constraints:** Prioritize user health and safety by designing ergonomic, user-friendly interfaces that minimize distractions and cognitive load, reducing the risk of accidents. The system should meet automotive safety standards and include accessibility features for all users, including those with disabilities.
- **Political Constraints:** Ensure compliance with relevant laws, regulations, and government policies related to transportation, data privacy, telecommunications, and vehicle safety. Consider regional variations in regulations to facilitate the system's deployment across different markets and jurisdictions.

Based on user needs and project constraints, the need-constraint matrix (Fig. 2) was created. Solutions are scored as follows: a '+' (score +1) if the solution meets the need and respects constraints, a '-' (score -1) if it fails to meet the need and violates constraints, and a '0' if it meets the need but violates constraints. The solution with the highest score is selected for further development. The matrix revealed that an audio-based navigation system is the optimal solution. This led to the creation of an app designed to serve as a verbal routing partner, providing confirmations, guidance, and affirmations as if a person were sitting beside you.

3. Prototyping

With the idea developed regarding what exactly needs to be made to help drivers with the problem, S.A.R.A (Smart Automated Route Assistant) was designed. Figure 3 summarizes the components of the system and how they are structured in relation to each other.

The flow of the process looks as follows: The user asks for something -> Wake word model checks if the user is actually speaking to the AI or is chatting with others -> AI gets activated and starts listening -> converts the recorded audio from the user to text -> sends the converted request to the fine-tuned LLM model -> Response is received and sent to a text-to-speech API for audio generation -> audio is received and played to the user as the AI's voice. This is an oversimplified workflow of how the request of the user is being processed behind the scenes. However, there's a lot more happening to provide the best guidance for the user, which was spared to give an intuitive idea of how the app's backbone is set up.

There are 6 broad components to the app, as illustrated by different colors in Fig. 3, each representing a module with subcomponents. They are: (1) The App Frontend/UI, (2) Video Processing, (3) NLP1, (4) API calls, (5) Algorithm, (6) NLP2.

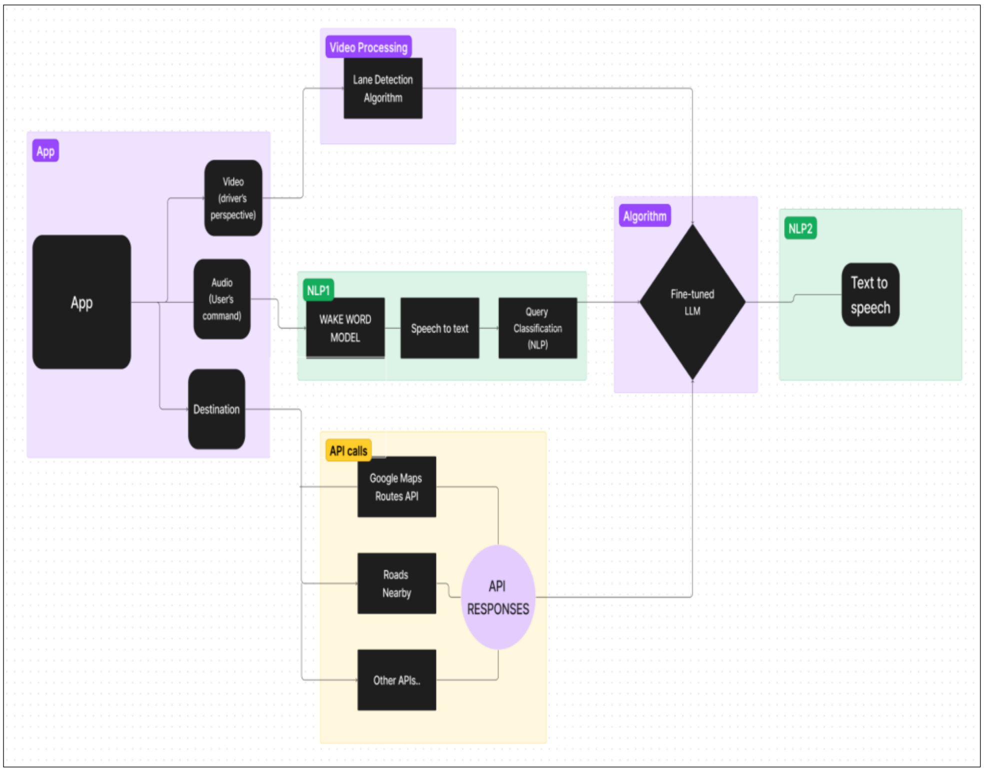


Figure 3. System architecture

Also, the overall design structure can be represented in the form of the input-process-output (IPO) model, whereby each component mentioned is the smallest unit of component present in the system. The IPO model is represented in Table 1.

Table 1: The Input-Process-Output (IPO) Model of the System

INPUT	PROCESS	OUTPUT
1) GPS Geolocation 2) GPS Navigation 3) Live Video Stream (Websockets Protocol) 4) Voice Prompt	1) Google Maps Navigation API 2) Speech-to-Text 3) Query Classifications 4) API calls 5) Fine-tuned GPT Model 6) Lane-detection Algorithm 7) Lane Designation Algorithm 8) Text-to-Speech	1) Navigation Interface 2) Audio-prompted Voice Assistance (Text-to-Speech)

The description of each of the components, their relation to the functionality of the app, and their testing are explained in the following section.

3.1. The App Frontend/ UI

The app's User Interface (UI) is designed to be familiar to users of popular GPS navigation apps, consisting of three main screens: the map view page, the search page, and the navigation page. The **map view page** displays the user's current location and provides a bird's-eye view of the route to their destination, with features like a search textbox and an AI indicator that shows when the AI is active and listening. It also offers an overview of the route with colored polylines and an estimated time of arrival. Clicking the search textbox directs users to the **search page**, where they can look up destinations using the Google Places API, with results biased towards nearby locations. Once a destination is selected, users return to the map view page for a trip summary and confirmation. Figure 4 displays the 3 pages: home, search, and navigation Page respectively.

The **navigation page** is the core of the UI, where users will spend most of their time. It visually guides them from their origin to their destination, similar to other GPS-based navigation apps like Google Maps and Waze. This screen is designed to be intuitive and user-friendly, ensuring a seamless navigation experience. The navigation map provides visual route guidance along with brief commands like "take right after 100m." As the driver approaches an intersection and needs lane guidance, they rely on the app's navigational AI assistant to clarify the correct lane.

3.2.1 NLP-1 Wake-Word Model

The app continuously records audio at a sample rate of 16000 Hz to efficiently manage the device's resources, switching to the device's default rate if unsupported. This audio is fed into a wake word model composed of a pre-processing function, a shared feature extraction backbone, and a classification model. The process starts with capturing audio in real-time, which is then transformed into a mel-spectrogram using the Fourier Transform and mel scale to highlight speech-relevant frequencies. This spectrogram is processed by a convolutional neural network (CNN) to create speech audio embeddings, which are then passed through a classification model to detect the presence of the wake word. The model was trained on the wake words "Sara", "Sarah", "Saara", "Saaraa", "Hey Sarah", "Hey Sara". Consequently, the user must first call "Sara", or any other variation from the above, and then proceed to give its query. The wake-word model used in the app is an adaptation of an open-sourced model, openWakeWord [23].

The testing involved three distinct scenarios to determine the model's performance. In **Test 1**, the model was run in an environment with background music to measure how often it falsely detected the wake word (false positives) without it being uttered. **Test 2** involved running the model in a quiet environment where the wake word was intentionally spoken 20 times, assessing its ability to correctly recognize the wake word without any background interference. **Test 3** combined elements of the previous tests, with the wake word spoken 20 times amidst background noise, to evaluate how well the model could distinguish the wake word in a more challenging, real-world scenario. The results of these tests were used to construct a confusion matrix, providing insights into the model's overall accuracy and precision.

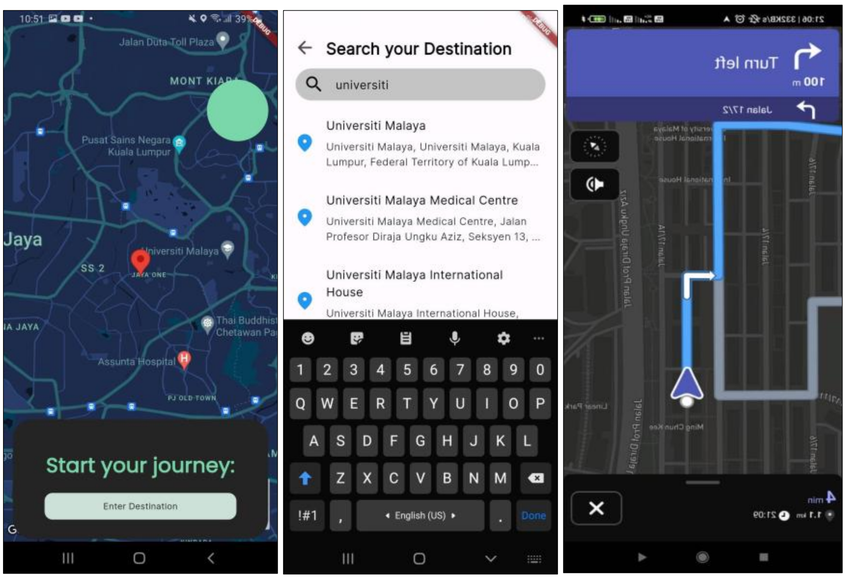

Jalan Duta Toll Plaza
MONT KIARA
Pusat Sains Negara Kuala Lumpur
Jaya
SS 2
Universiti Malaya
JAYA ONE
Thai Buddhist Chetawan Pa
Assunta Hospital
PJ OLD TOWN
Start your journey:
Enter Destination
Search your Destination
universiti
Universiti Malaya
Universiti Malaya, Universiti Malaya, Kuala Lumpur, Federal Territory of Kuala Lump...
Universiti Malaya Medical Centre
Universiti Malaya Medical Centre, Jalan Profesor Diraja Ungku Aziz, Seksyen 13, ...
Universiti Malaya International House
Universiti Malaya International House,
English (US)
Done
Turn left
100 m
Jalan 17/2
4 min
1.1 km
21:09

Figure 4. Home Page, Search Page, and Navigation page

3.1.1.1. *Testing and Results*

The wake-word model's evaluation involved determining its prediction timing and accuracy through a series of defined tests. First, the model was run for 10 seconds (t) to count the number of predictions (p) made, with the time per prediction (T) calculated using Equation (1).

$$T = p/t. \qquad (1)$$

The model was tested over a 20-second period, yielding 20 predictions. For wake-word tests, each utterance lasted 1 second. In Test 1, predictions under 0.6 were True Negatives, and those over 0.6 were False Positives. In Test 2, with 20 wake-word utterances, predictions over 0.6 were True Positives, and those under 0.6 were False Negatives. In Test 3, with 40 wake-word utterances in a noisy environment, the total predictions were calculated using Equation (2).

$$N = td \ * T \qquad (2)$$

where *td* is the duration of the test. Predictions during utterances were classified as True Positives or False Negatives, while predictions from background noise

$$Nf = N - 40 \qquad (3)$$

were classified as True Negatives or False Positives. Finally, confusion matrices were constructed from the results of Tests 1 and 2 combined shown in Table 2, and another from Test 3 alone shown in Table 3. The accuracy of the model is, therefore, (15+16)/40 = 0.775 or 77.5%. The precision of the model is, therefore, (15/15+4) = 0.789 or 78.9%. The recall of the model is, therefore, (15/15+5) = 0.75 or 75%. The F1 score, described as the harmonic mean of precision and recall, making it high only when both values are high, is a good measure to evaluate a model with imbalanced datasets. It is given by Equation (4).

$$F1\ Score\ =\ 2 * \frac{Precision*Recall}{Precision+Recall} \tag{4}$$

The F1 score of the model is 2*0.789*0.75/(0.789+0.75) = 0.769 or 76.9%.

Table 2. Confusion Matrix of Test-1 & Test-2 combined**.**

Test-1 & Test-2 Results N= 20+20		Actual	
		Positive	Negative
Predicted	Positive	*tp:15*	*fp: 4*
	Negative	*fn: 5*	*tn: 16*

Table 3. Confusion Matrix of Test-3

Test-3 Results N=52, NF = 12		Actual	
		Positive	Negative
Predicted	Positive	*tp:32*	*fp: 3*
	Negative	*fn: 8*	*tn: 9*

From the data in Table 3, the accuracy of the model is, therefore, (32+9)/52 = 0.788 or 78.8%. The precision of the model is, therefore, (32/32+3) = 0.914 or 91.4%. Recall of the model is 32/(32+8) = 0.8 or 80%. Since the data is imbalanced for test-3, F1 score becomes a good metric to judge the model. Therefore, the F1 score of the model is: 2*0.914*0.8/(0.914+0.8) = 0.853 or 85.3%. From the above testing, it can be inferred that the model shows a strong performance in general with high precision and a good balance between precision and recall, as reflected by the F1 score. To continue with the system design explanation, we assume that the user has now called the AI assistant by invoking the wake word, “Sara”, and the model has detected it and “woken up”.

3.1.2. Speech to Text

The next stage in the pipeline after detecting the wake word is listening to the user’s query and converting it into text. The app listens to the user’s query and converts it to text using built-in speech-to-text models on iOS and Android. Built-in models are chosen for offline use and reduced latency, though future updates might switch to Whisper (from OpenAI) for its higher accuracy and local execution. For the prototype, the speech_to_text package from the flutter library was used to convert speech to text, The library uses the native API of android and iOS platforms to perform the speech-to-text conversion. The model works best only for English, which is good enough for the prototype. Since this is a standard library, no further testing was done on it as it’s metrics are known [9].

3.1.3. Query Classification

The system should be able to detect what type of query is being asked so that it can bring up the necessary data to guide the driver. Based on survey results, four key functionalities were identified: Lane Guidance, Contextual Guidance, Traffic Data Assistance, and Hands-Free Functionality. To address these, we developed a text classification model using DistilBERT, a lightweight version of BERT [17]. The model processes input IDs and attention masks, generating hidden states and a pooled output through a dense layer with ReLU activation and a dropout layer to prevent overfitting. A final dense layer with softmax activation performs multi-class classification. The model, trained on 1,238 English queries using categorical cross-entropy loss and the Adam optimizer, contains approximately 66 million parameters.
The model’s performance was tested on a set of 250 queries, equally representing each class. Due to the limited dataset size, the model achieved exceptionally high accuracy, precision, and F1-scores of 99% across all metrics. Despite the absence of an ablation test, the model effectively classified user queries related to lane guidance, contextual info, traffic data, and hands-free functionalities, enabling seamless activation of relevant APIs for traffic and navigational data retrieval.

3.2. Video Processing

To address drivers' lane navigation issues, a lane detection system was developed. The system uses a custom API to detect and index lanes from video frames (resolution ≤ 720p), processed by a PyTorch model hosted on a cloud server in Singapore. The model, based on [10][11][12], incorporates a backbone network, localization branch, existence branch, and a mathematical expectation module for accurate lane detection. Upon detecting the wake word, the app establishes a websocket connection to the backend to start real-time video streaming, avoiding continuous streaming to reduce data usage. The lane detection model employs a hybrid anchor system and ordinal classification for efficiency. The processed frames yield JSON data with lane count and indexing, facilitating real-time driver assistance.

3.2.3. Testing

A comparative test evaluated two algorithms for lane-related tasks: "Ultra Fast Lane Detection" for lane detection and another for lane indexing. The first algorithm was reportedly tested using live video streams from YouTube, simulating real-world conditions efficiently. The second algorithm was tested by uploading MP4 files to a server via HTTP POST requests, and receiving JSON responses. It was assumed that using WebSockets would yield similar results. Figure 5 and Fig. 6 given an idea about the results.

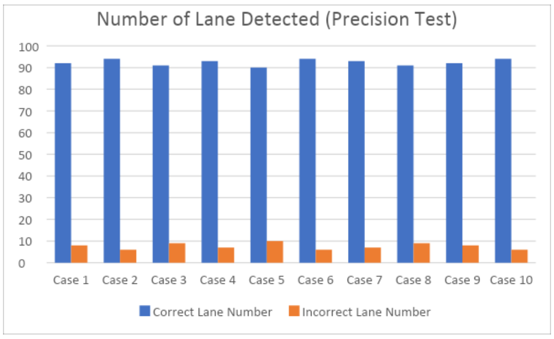


Figure 5. Test on number of lanes detected

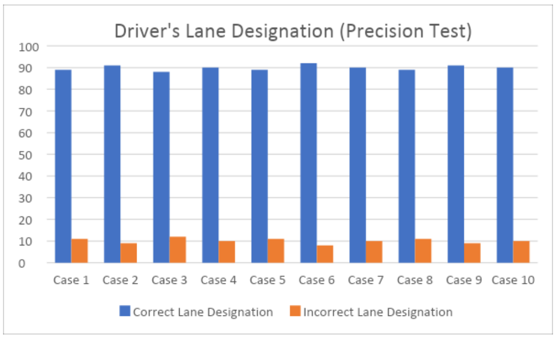


Figure 6. Test on driver Lane designation

Both lane detection and lane designation showed high precision, with detections exceeding 90% and designations averaging 89%. Lane detection consistently achieved around 92% precision, while lane designation had slight variations across tests. However, a noticeable discrepancy arises in the incorrect lane designations and detections. This discrepancy is primarily attributed to the app's occasional failure to register lanes from other directions as non-related to the driver's lane. This misclassification results in a small but significant number of incorrect counts. Such errors typically occur in complex traffic scenarios where the app struggles to differentiate between relevant and non-relevant lanes, leading to these inaccuracies. The video processing API and other external APIs provide data to the fine-tuned LLM, which then offers detailed navigational guidance to users.

3.3. Fine-Tuned LLM

The Fine-tuned LLM, based on OpenAI's GPT-3.5 Turbo model, is central to this app. Fine-tuning refines a pre-existing model to specialize in a specific domain, such as navigation assistance. This process adapts the model from general text data to handle specific queries about routes, directions, and landmarks more effectively. By doing so, the app can give only the necessary, relevant and concise responses.

For this app, the LLM was trained on 150 sample conversations (sample conversation can be seen from the appendix, section 1.1), which included system messages setting the assistant's behavior, user queries, and assistant responses. This fine-tuning improves the model's ability to provide detailed and relevant navigational guidance.

3.3.3. Testing

To evaluate the fine-tuned model's effectiveness, a subjective test was conducted using Google Forms. Participants rated the LLM's responses on a scale of 1-5, with 1 being "very unclear" and 5 "very clear." Although only 14 participants completed the survey, the feedback provided useful insights into the model's clarity and contextual relevance. The results suggest that, overall, the instructions are clear and can be understood clearly. The summary of the results can be viewed from the appendix (section 1.2). The response of the fine-tuned LLM must now be converted back to speech for the user to hear the guidance instruction.

3.4. NLP 2: Text to speech

To convert the text back to speech so that the driver can listen is done using OpenAi's text to speech model. OpenAi have made an endpoint available that allows developers to access their state-of-the-art text-to-speech (TTS) model. The main reason for using this TTS model is because it produces highly natural-sounding speech [13].

3.5. Routing Data Collection – API calls

For the prototype, to get the necessary routing and directions data, the APIs mentioned in Table 4 are used. Their respective use in the app is also mentioned.

Table 4. APIs used.

API	Use
Places, Places Nearby (Google Maps)	For searching up the location and retrieving contextual info on the route of the user respectively [18][19]
Roads Nearby (Google Maps)	For efficient lane guidance; to let users know if it's the upcoming turn or the turn ahead [20].
Traffic API (TOMTOM)	To supply users with the latest traffic info on their route [21].
Routes API (Google Maps)	To determine the route from the users's location to their destination [22].

3.6. Integration & Testing

The app integrates various components, starting with the smooth incorporation of the speech-to-text functionality using Flutter's speech_to_text package, which accurately converts spoken words into text. Integrating the text classification model revealed a challenge: the lack of a preprocessing layer prevented direct input of text data. This issue is currently addressed by focusing on lane guidance queries only. API calls via Flutter's http package and the fine-tuned GPT model for navigational queries were seamlessly integrated. For live video streaming, we transitioned from HTTP POST requests to WebSockets for real-time communication, enhancing performance. Lastly, the text-to-speech functionality was implemented using OpenAI's model, overcoming initial issues with MPEG audio format by switching to .opus, and adjusting for real-time playback by first saving data to a file.

The primary concern regarding the app's usefulness is its response time. While the AI's adequacy has been tested, this section focuses on latency. Latency, which measures the time taken for various app operations, is crucial for user experience, ensuring quick responses. Key actions such as entering a destination, text classification, API calls, and audio prompts were timed in milliseconds to assess responsiveness. Although on-road latency tests were not conducted, multiple prompts were tested with video processing disabled. The results are summarized in Table 5.

Table 5. Latency tests summary

Operation	**Case 1**	**Case 2**	**Case 3**	**Case 4**	**Case 5**	**Case 6**	**Case 7**	**Case 8**	**Case 9**	**Case 10**	**Avg (Ms)**
Audio response Latency	259	301	249	256	170	260	329	256	184	190	255.55
API Call Latency	185	173	182	173	180	178	179	181	176	179	179.48

From the results, it’s seen that the response times fluctuate greatly; sometimes the responses are quick enough whilst other times it can be a bit delayed. Although the average latency response of 255.55 milliseconds should be good enough to respond to the drivers in time, the effectiveness of it cannot be validated as of now. Response times below 300 ms are generally considered satisfactory for user experience [14][15]. However, for driving and navigation the case may be different. Here is an opportunity for future works to determine if the given response time is effective to give timely responses.

4. Inference and Conclusion

Yang *et al*. [14] investigated user acceptance of navigation apps, incorporating factors such as drivers’ sense of direction, navigation application affinity, and distraction perception. Their findings indicated that higher distraction reduced navigation application affinity, which in turn affected drivers’ perception of the app’s usefulness. The aim of developing this app was to enhance usability, especially in preventing wrong-way driving (WWD) by allowing drivers to focus on the road while interacting with the app through voice commands. Although survey results suggest strong user interest, further testing and on-road trials are necessary to assess the app’s effectiveness in addressing WWD. Ref. [5] highlights limitations regarding navigation app guidance distances, which are difficult to compare since our study measures response speed in time, not distance. Future enhancements will address these issues. To conclude, this project addresses the critical issue of distracted driving caused by GPS systems. The solution is an advanced audio-based navigation system utilizing AI to enhance safety and reliability. It provides clear, context-sensitive instructions, reducing the need for drivers to look at screens and encouraging them to focus more on the road. Key achievements of this project include:

- Reduction of Cognitive Overload: It means that our AI assistant gives clear and timely audio directions which lessens the load of interpreting the visuals given to the user.
- Enhanced Driver Focus: In this way, a driver does not have to constantly check what is displayed on a screen, thus preventing the accident potentially happening.
- User-Friendly Interaction: The system is based on the speech recognition and natural language processing to provide the driver with convenient interaction with the car. The drivers are able to ask questions, request clarification and additional information and get real time feedback without necessarily having to hold up the interaction for it to go through a human element.
- Real-Time Adaptability: While using GPS and accelerometers, the system is able to give real time information on the route being taken by the vehicle and the route changes to fit the road condition and driver behavior due to integration of Google Maps API.

References

[1] M. A. Butorac and J. C. Wen, *Access Management on Crossroads in the Vicinity of Interchanges*, Project 20-5 FY 2000, Transportation Research Board, Washington, DC, 2004.

[2] F. Baratian-Ghorghi, H. Zhou, M. Jalayer, *et al.*, "Prediction of potential wrong-way entries at exit ramps of signalized partial cloverleaf interchanges," *Journal of Traffic Injury Prevention*, vol. 16, no. 6, pp. 599-604, 2015.

[3] National Highway Traffic Safety Administration (NHTSA), *Fatality Analysis Reporting System (FARS) Encyclopedia*. Available: http://www-fars.nhtsa.dot.gov/main/index.aspx. Accessed: May 1, 2016.

[4] D. A. Morena and T. J. Leix, "Where these drivers went wrong," *Public Roads*, vol. 75, no. 6, pp. 33-41, 2012.

[5] M. Jalayer, H. Zhou, and B. Zhang, "Evaluation of navigation performances of GPS devices near interchange area pertaining to wrong-way driving," *Journal of Traffic and Transportation Engineering (English Edition)*, vol. 3, no. 6, pp. 593-601, 2016. doi: 10.1016/j.jtte.2016.07.003.

[6] Chen, C., Li, C., Zhao, H., Gan, C., & Schissler, C. (2020). SoundSpaces: Audio-visual navigation in 3D environments. In A. Vedaldi, H. Bischof, T. Brox, & J. M. Frahm (Eds.), Computer Vision – ECCV 2020. Lecture Notes in Computer Science (Vol. 12351). Springer. https://doi.org/10.1007/978-3-030-58539-6_2

[7] Zhao, S., Zhou, Y., & Huang, T. (2022). A novel method for AI-assisted INS/GNSS navigation system based on CNN-GRU and CKF during GNSS outage. Remote Sensing, 14(18), 4494. https://doi.org/10.3390/rs14184494

[8] Khan, M. A., Paul, P., Rashid, M., Hossain, M., & Ahad, M. A. R. (2020). An AI-based visual aid with integrated reading assistant for the completely blind. IEEE Transactions on Human-Machine Systems, 50(6), 507-517. https://doi.org/10.1109/THMS.2020.3027534.

[9] "Ultra-Fast Lane Detection v2," cfzd, GitHub, [Online]. Available: https://github.com/cfzd/Ultra-Fast-Lane-Detection-v2. [Accessed: Aug. 31, 2024].

[10] Udacity, "Advanced Lane Finding: Self-Driving Car Engineer Nanodegree Program," GitHub repository, [Online]. Available: https://github.com/udacity/CarND-Advanced-Lane-Lines

[11] Z. Qin, H. Wang, Y. Wang, and Y. Yang, "Ultra Fast Deep Lane Detection with Hybrid Anchor Driven Ordinal Classification," GitHub repository, [Online]. Available: https://github.com/cfzd/Ultra-Fast-Lane-Detection-v2

A. Fox, S. D. Gribble, E. A. Brewer, and E. Amir, "Adapting to network and client variability via on-demand dynamic distillation," in *ACM SIGPLAN Notices*, vol. 31, no. 9, pp. 160-170, Sept. 1996.

[12] LITSLINK. "Top 7 AI App Ideas Using OpenAI's TTS Model." Accessed: Aug. 31, 2024. [Online]. Available: https://litslink.com/blog/app-ideas-openais-new-tts.

[13] L. Yang, Y. Bian, X. Zhao, X. Liu, and X. Yao, "Drivers' acceptance of mobile navigation applications: An extended technology acceptance model considering drivers' sense of direction, navigation application affinity and distraction perception," *Int. J. Hum.-Comput. Stud.*, vol. 145, p. 102507, 2021. [Online]. Available: https://doi.org/10.1016/j.ijhcs.2020.102507.

[14] Open Voice Network, "Voice Assistant Clarity Standards," The Linux Foundation, 2023. Available: https://www.linuxfoundation.org/

[15] J. Nielsen, "Usability Engineering," Academic Press, 1993.

[16] Hugging Face, "DistilBERT base uncased," https://huggingface.co/distilbert/distilbert-base-uncased. Accessed: June 4, 2024.

[17] Google, "Google Maps Places API Overview," https://developers.google.com/maps/documentation/places/web-service/

[18] Google, "Google Maps Places API Nearby Search," https://developers.google.com/maps/documentation/places/web-service/search-nearby?hl=de. Accessed: April 11, 2024.

[19] Google, "Google Maps Roads API Overview," https://developers.google.com/maps/documentation/roads/overview?hl=de. Accessed: April 11, 2024.

[20] TomTom, "Traffic APIs," https://www.tomtom.com/products/traffic-apis/. Accessed: April 11, 2024.

[21] Google Cloud, "Routes API," https://console.cloud.google.com/apis/library/routes.googleapis.com?hl=en-GB&pli=1. Accessed: Nov. 11, 2024.

[22] D. Scripka, *openWakeWord*, GitHub repository, [Online]. Available: https://github.com/dscripka/openWakeWord. [Accessed: Apr. 28, 2024].

[23] M. Jalayer, H. Zhou, and B. Zhang, "Evaluation of navigation performances of GPS devices near interchange area pertaining to wrong-way driving," *Journal of Traffic and Transportation Engineering (English Edition)*, vol. 3, no. 6, pp. 593-601, 2016. [Online]. Available: https://ars.els-cdn.com/content/image/1-s2.0-S209575641630109X-gr3_lrg.jpg.

Appendices

1.1. Sample conversation between driver and AI (SARA):

User: Hi there.. Can you help me navigate from Jalan 19/22 ?
AI Assistant: Heyy! From Jalan 19/22, you'll want to head west toward Jalan 19/21.
User: Which lane should I be in for that turn?
AI Assistant: You'll want to be in the left lane for that turn.
User: Okay, got it. After that turn, which direction should I take?
AI Assistant: After turning left onto Jalan SS 2/72, you'll need to be in the right lane to prepare for a right turn onto Jalan SS 2/95.
User: Gotcha. And after I make that right turn, what's next?
AI Assistant: You'll continue straight on Jalan SS 2/95 for about 0.5 km, then take a left turn at Jalan SS 2/2.
User: Any good food spots along the way?
AI Assistant: There's "my Burger Lab Seapark" ahead. They serve burgers.
User: Awesome, thanks! After grabbing a bite, I might need to stop by a bank. Any nearby?
AI Assistant:You'll find Maybank (SS2 PJ) not far from "my Burger Lab Seapark". Hope you enjoy your burger there!

1.2. Summary of the results from the subjective test on the fine-tuned LLM model:

Question	Summary and Inference
Question 1: The instruction "You will see three exits ahead. Please take the second exit which is right after Smith's Cafe"	The instruction was rated positively, with the majority of respondents giving it a '4' or '5' for clarity. This indicates that the instruction was generally clear, though there is some room for improvement as a few respondents rated it lower.
Question 2: The instruction "Just before the exit, you'll see Eastin Hotel on your left side. Once you see this hotel, start moving to the right-most lane to take your exit"	The instruction received predominantly positive ratings, with the majority of respondents rating it as '4' or '5' for clarity. This indicates that the instruction was generally well-understood by most participants.
Question 3: The instruction "Once you pass the Chemistry Department, continue for about 20 seconds and take the next right onto Lingkaran Wawasan"	The instruction was generally perceived as clear, with 78.6% of respondents rating it as '4' or '5'. No respondents found the instruction completely unclear, but 21.4% rated it as '3', indicating some potential for improvement in clarity.
Question 4: The instruction "After you turn left, continue for approximately 25 seconds. Then, you'll make another left onto Jalan 16/4 just after Luzini furniture store"	The instruction received mixed ratings, with 35.7% of respondents rating it as '3' and an equal percentage rating it as '5'. This indicates that while a significant portion of respondents found the instruction very clear, there is a notable portion that found it moderately clear, suggesting some room for improvement in clarity.
Question 5: The instruction "You will need to prepare to turn left shortly. Please move to the left lane as you approach the intersection at Lingkungan Budi"	The instruction was rated highly, with 92.9% of respondents giving it a '4' or '5'. This indicates that the majority of participants found the instruction clear and easy to understand, with only one respondent rating it as '2', suggesting a slight area for improvement.
Question 6: The instruction "The turn is approximately 300 meters ahead, around 30 seconds. You will see the road signage as you approach the turn"	The instruction was generally well-received, with 78.6% of respondents rating it as '4' or '5'. This indicates that the majority of participants found the instruction clear, although 21.4% rated it as '3', suggesting a small opportunity for further clarification.
Question 7: The instruction "There are 4 lanes ahead, stay on the second-most lane from right"	The instruction was rated very highly for clarity, with 71.4% of respondents giving it a '5' and 14.3% giving it a '4'. This indicates that the majority of participants found the instruction very clear, with only a small portion (7.1%) finding it moderately clear. There were no responses indicating confusion or lack of clarity.

Question 8: The instruction "Look for the Senada Residences on your right as you get closer to the intersection. It's a prominent landmark nearby"	The instruction was well-received, with 71.4% of respondents rating it as '4' or '5'. This indicates that the majority of participants found the instruction clear, although 28.5% rated it as '2' or '3', suggesting that there is some room for improvement to increase its clarity for all users.
Question 9: The instruction "You can avoid the traffic delay by taking Second Avenue instead, which will reconnect you to First Avenue without significant traffic issues. Just take the right turn in 20 seconds"	The instruction received mixed ratings. While 57.1% of respondents rated it as '4' or '5', indicating clarity, a significant portion rated it as '3' (28.6%), and there were some low ratings (7.1% each for '1' and '2'). This suggests that while more than half of the participants found the instruction clear, there is notable room for improvement to enhance its clarity for a broader audience.
Question 10: The instruction "There is a reported incident on the route from Second Avenue to Third Boulevard. It involves a delay of about two minutes over a stretch of 100 meters. The incident is expected to last until 1:00 PM today"	The instruction was generally considered clear, with 85.8% of respondents rating it as '4' or '5'. This suggests that the majority of participants found the instruction to be clear and informative. The remaining 14.3% rated it as '3', indicating that while it was mostly clear, there is a slight opportunity for improvement.

1.3. Problem Validation survey results

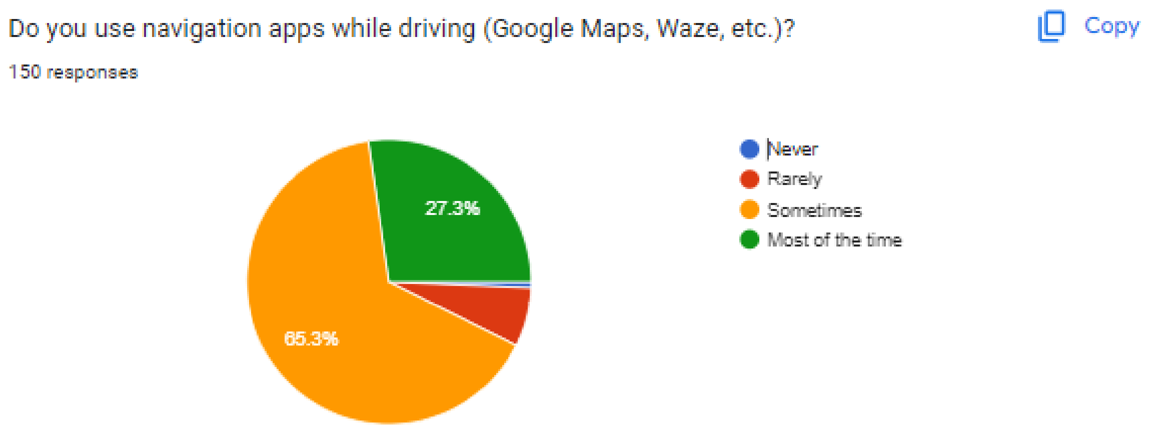
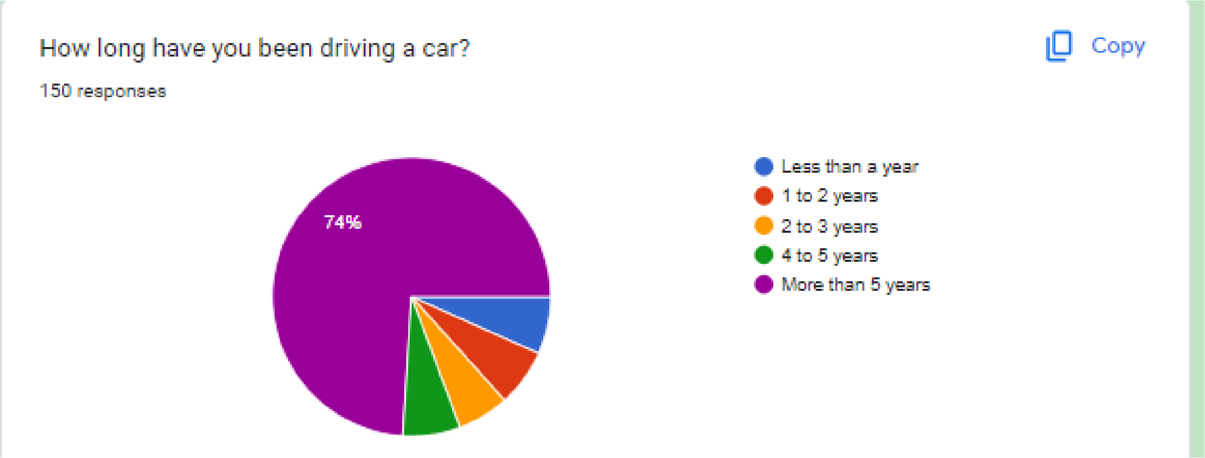
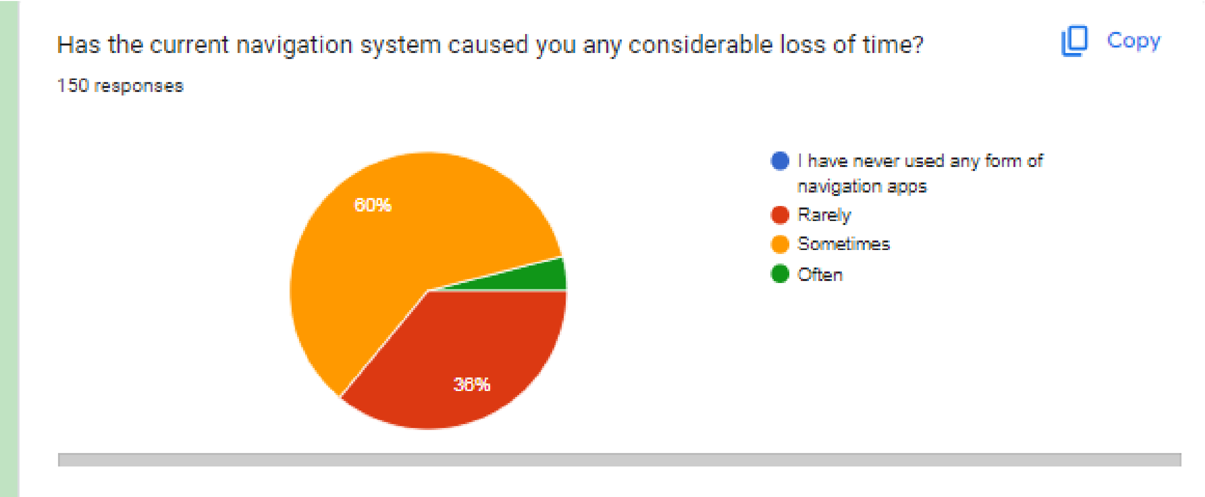


Figure 7. The data from google form that showcases the percentage of drivers that uses navigation applications


How long have you been driving a car?

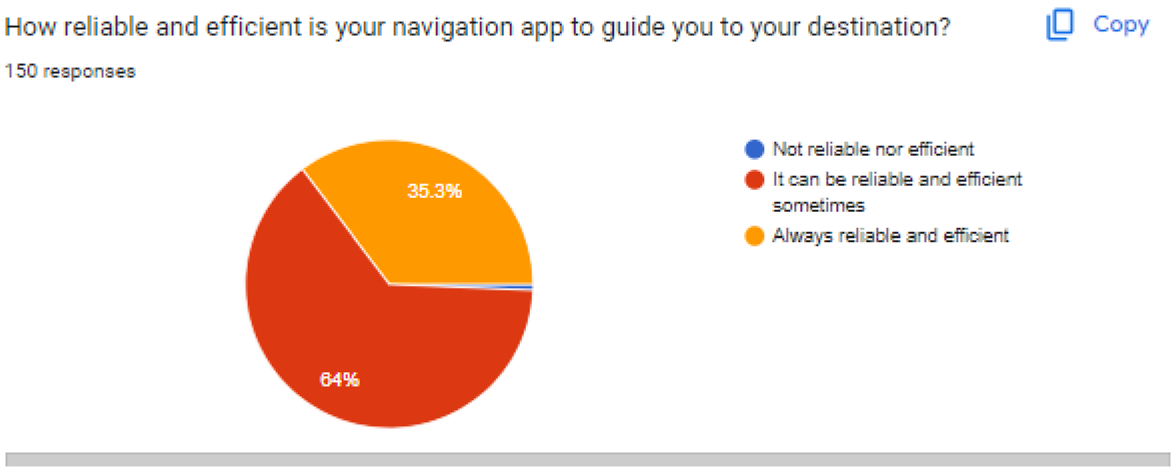
150 responses

Figure 8. The data from the google form that showcases the percentage of how long users have been driving a car.


Has the current navigation system caused you any considerable loss of time?

150 responses

Figure 9. The data from the google form that showcases the percentage of whether or not the current navigation system has caused any considerable loss of time to drivers.



How reliable and efficient is your navigation app to guide you to your destination?

150 responses

Figure 10. The data from the google form that showcases the percentage of how reliable and efficient the current navigation application is in guiding drivers to their final destination.