Plus Go: Intelligent Complementary Ride-Sharing System

Viraj Wickramasinghe[1], Ashane Edirisinghe[2], Surath Gunawardena[3], Athrie Gunathilake[4], Janaka Wijekoon[5]
Department of Software Engineering
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
1.vjanuradhawick@gmail.com 2.ashane.edirisinghe@gmail.com 3.surathruwan@gmail.com 4.athrienathasha@gmail.com
5.janaka.w@sliit.lk

Abstract*—Traffic congestion has drawn the attention of the society as a dominant concern. The major problem is the number of vehicles daily entering the urban areas a high. Most of the time, many vehicles are coming from the same area to a defined destination, resulting the vehicles are underutilized merely because those drivers are used to travel alone. To reduce this, the current study proposes a solution to minimize this problem by implementing a ride-sharing platform: Plus Go. An initial study of the Plus Go platform has targeted the office staff in Sri Lanka where the majority of the offices are operated in highly congested urban areas. Plus Go does outperform other existing ride-sharing platforms as it matches the preferences and professional level of the passengers with the drivers, and vice versa; next recommends the rides by utilizing trajectory details, gender preference, rating, time and date as the parameters. Furthermore, the Plus Go platform uses computer vision methodologies to avoid fake registrations, and nonetheless, it uses crowd-sourcing platforms to increase the accuracy of the suggested routes. The proposed platform demands the passengers, including the driver to share the fare for each ride session; intelligently estimated based on fuel consumption of the vehicle, distance, the time taken to reach the destination, are taken as the parameters for it. This estimation is further optimized at the end of each ride-sharing session using the statistics elicited from the OBD2 scanner. Plus Go maintains a customized rating system which proffers the users an opportunity to rate and write reviews regarding the ride-sharing session; these reviews are probed with a sentiment analysis algorithm to optimize the ratings estimated and resulting the users to have more reliable ride-sharing experience.

Keywords—Ridesharing, Ride-matching, Machine Learning, Image Processing, Crowdsourcing

I. INTRODUCTION

According to the Report of World Population Prospects 2017 in the United Nations, by 2030, the Sri Lankan population forecast will be around 21,474,701. Hence, the demand for private transportation has also increased over the last few years[1]. Corresponding to the articles, surveys, and police reports, there is a gradual increase in vehicles that enter to Colombo during rush hours[3]. As a result of that vast traffic congestion has occurred in urban cities like Colombo city during office hours. Because of the increase in traffic congestion, the number of accidents also increases. According to the surveys conducted by the Central Bank, they have identified low occupant vehicles like cars that have been in high quantity and resulted in traffic congestion. Hence, this is a waste of financial resources, moreover polluting the environment severely. By considering some statistics collected from articles and reports, we identified that traffic is a significant concern in urban areas, especially during office hours. It has been stated that there is an Rs.500 million loss due to the daily traffic congestion in Sri Lanka[3]. Due to the heavy traffic in the rush hours, there is a massive loss of time wasted on the road. Therefore we came up with a concept of ridesharing application as a solution for this huge problem. The basic idea behind that was to combine professionals who are traveling to work by their private vehicles. This will lead to building the network among professionals. So this proposed ridesharing application will help to reduce stress and improve productivity while traveling as a passenger.

Hence, when one vehicle carries several numbers of people together, it will reduce the number of cars entering the city. To verify our hypothesis, we surveyed a sample of more than 150 office crowd, and we identified that the majority considered ridesharing as an excellent solution to reduce traffic in urban areas. Apart from the traffic minimization, this will provide some other traveling facilities to the users. When single occupants share their ride among other occupants, who go to the same destination, which helps to reduce the number of vehicles on the road which results in reducing the traffic congestion and minimize the emission of carbon dioxide (CO_2) to the environment, consequently, CO_2 emissions (metric tons per capita) in Sri Lanka was reported at 0.88555 in 2014 can be drastically reduced from this[11]. Resultantly it will minimize environmental pollution too.

Furthermore, we surveyed to get the real-time fuel consumption of the vehicles in the urban areas[2]. Our proposed system will predict the ride fare of the passenger before the ride starts and help to reduce the cost of traveling by distributing the total fare among all the passengers.

II. LITERATURE REVIEW

Ridesharing is considered as a possible solution to reduce traffic congestion[4]. Factors like cost of travelling, time, distance, ownership of the vehicle, personal preferences influenced in choosing the individual transportation mode[14, 15]. Certain studies have focused on the factors like fuel price, demography and safety measures in selecting the transport mode[16, 17]. Manzini and Pareschi[5] introduced a new Decision Support System (DSS) for ridesharing systems. Passengers were given support to select the best car to be travelled. With our proposed method, we optimize the solutions provided by the previous studies by suggesting only the drivers, which match the preferences and professional level of the passenger.

Swati, Neha, and Ajita[6] did another research to make the ridesharing application user-friendly of both the driver and the passenger. From the point of algorithms, Fagin and Williams(1983) and later Ajtai et al. (1998)[7] proposed the algorithm to get a portion of users that willing to share their cars and from that study, they introduced a new scheduling algorithm. For the process of grouping individuals who matches with each other, was first done through using K-Means Clustering. Abdul and Sebastian[8] proved that the mean accuracy of using K-Means clustering gives you over 78% accuracy depending on the dataset, we are using. They also enhanced some features in the K-Means and proved that the new Enhanced version of the K-Means yield accuracy over 89%. From the study done by Chih-Ping Wei, Yen-Hsien Lee and Che-Ming Hsu to compare the most accurate clustering algorithm, they have proved that K-Means clustering yields precise comparatively results than the other clustering algorithms they used in the study[9]. From the literature, the number of clusters in a dataset is easily calculated using the elbow method from the study of Ketchen. and Shook[10], they have analyzed Elbow method returns accurate value depending on the dataset used. In this study, we will be using Elbow method to calculate the number of clusters and K-Means clustering to cluster the drivers and passengers.

To validate the images of NIC and License, it is needed to proceed with image processing. Valiente, Sadaike, Gutiérrez, Soriano, Bressan, and Ruggiero(2016) explains that uploading the images to a server and do the manipulations there is an optimum method[11]. This image manipulation process could be done by following the steps as loading the image, cropping Image, processing the image and then extracting information from the image[18, 19]. In the way, Chakraborty and Mallik(2013) explain, the colour images have to be converted to grayscale and then to binary using an adaptive threshold for an accurate recognition[20, 21]. According to Mordvintsev and Abid(2017)[12], image processing could be easily performed with the help of OpenCV library. It also provides the capability for face detection using Haar Cascade. With reference to Clark(2018)[13], the Python Imaging Library (PIL) has the ability to add image processing functionalities to the python interpreter, and it could be used in image archives, image display, and image processing. It was also identified that the processed images could be sent for recognition using Tesseract recognition engine which is considered as the best highly portable open source OCR engine currently available[18, 20, 22, 23].

Naive Bayes classifier is a simple method based on the Bayes rule which assumes that the presence of a particular feature independent of the presence of any other feature and contributes independently to the final probability[25, 26]. The literature says that an advantage of Naïve Bayes' is that it only needs a little amount of training data for the estimation of the parameters required in classification[24]. Further, it has been identified that the Naïve Bayes algorithm performs better in sentiment analysis when it is compared with algorithms such as Support Vector Machine (SVM) and Multinomial Logistic Regression[24, 27]. According to the comparison between Naive Bayes, K-Nearest Neighbour and Random Forest Algorithms did by Baid, Gupta, and Chaplot(2017); suggests that the Naïve Bayes algorithm gave the best accuracy with the accuracy of 81.4%, while others give the accuracy of 55.3% and 78.65% respectively[25]. Further, Vidushi and Sodhi(2017) have emphasized that the Naïve Bayes algorithm outperforms the KNN algorithm[28]. By considering all the facts in the literature, we decided to use Naive Bayes in analyzing the reviews given by the users in our ride-sharing platform.

Since the general objective of this research is to minimize traffic congestion in urban areas, the identification of the optimum path is very significant. Dijkstra algorithm is known as the shortest path algorithm. Dijkstra Algorithm is used to identify the shortest path of the tree by considering the root of the tree as the starting point and then expand the tree node-by-node[29]. According to the weighted directed graph, there is the shortest path node. For the matter of fact, that node is starting from the starting point and extend to the earliest smallest point. In here, the point where entire nodes are adjacent to, is known as the smallest point, and the length of the arc is called the chord length[31]. According to this weighted directed graph, vertices of the graph emphasize the cities while the edges of the graph show the distances between two cities which adhere by a road. Therefore, Dijkstra's algorithm can be used to identify the closest path between a city and other entire cities[29, 30].

Similarly, Travelling Salesperson Problem (TSP) is another algorithm to identify the least weight tour, which covers all the nodes of the graph. According to the algorithm, it will identify a certain unique cycle in that graph and travel through each node of the graph for once with a minimal price[32]. Some algorithms have been developed to solve TSP. Adewole(2011) have been proposed Genetic Algorithm for solving Traveling Salesman Problem[33]. From the study of Dweepna Garg and Saurabh Shah(2011), they used some other method to solve Travelling Salesperson Problem, which is known as Ant Colony Optimization[34]. Tunon and Lopez (2005) proposed the algorithm of Branch and Bound to solve Travelling Salesman Problems[35]. Later on, Pragya, Dutta and Pratyush(2015) succeeded to identify another method to solve the Travelling Salesperson Problem, which is known as Dimensional Ant Colony Optimization (DACO)[36]. Find the shortest route by using the Travelling Salesman Problem based on Dijkstra's algorithm is more accurate since the algorithm extricates the distance from Google Maps [37, 38].

The fare calculation is considered as the business logic of this system. In a previous study done by Zoepf, Chen, Adu, and Pozo(2018), it was identified that the fare has to be calculated according to the vehicle type along with a minimum cost. Furthermore, it is based on time and distance added to the fare as dynamic factors. The driver has to bear all the expenses associated with the vehicle operation including depreciation, insurance, maintenance, repairs, and fuel which varies from driver to driver, or from vehicle to vehicle[39]. From the study of Santos and Xavier(2013), riders can decide how much the passenger is willing to pay for the ride. Then the system computes the cost based on the current fuel price and fuel consumption for the ride. Next, the system suggests a driver according to the willingness to pay the amount[40]. According to a study of Riquelme, Banerjee, and Johari, they identified two methodology for pricing .those methodologies are called as static and dynamic pricing. The static pricing method assigns a fixed amount for all the drivers who are in the platform. Hence, the price does not change based on the instantaneously available service capacity. These parameters are slightly different across the day (Even most taxicab services price in the evenings are different from time to time). Importance of the static price is that it doesn't affect to the instantaneous state, but it will only react to the course changes. They have derived an equation for dynamic pricing where their equation enters the list of available drivers, base price, distance and time and as a result, the dynamic price is calculated[41]. Considering the above literature regarding the fare calculation, there is no fixed way to calculate the fare for a single passenger. In our proposed solution, we will be using Multiple Linear Regression method to predict the fuel consumption, which results in estimating the fare for the ride. From the study of Aleksandar, Silvana and Valentina(2015), they have proved Mean Absolute Percentage Error(MAPE) of using Multiple Linear Regression in their study is 3.0730601 using a trained set of data. From that we have identified the error percentage of using regression model in critical analysis is comparatively lower than the other regression models and we'll be using it for our fare calculation procedure to produce an accurate result [42].

III. METHODOLOGY

Finding a mechanism to minimize traffic congestion in urban areas, especially in Colombo is considered to be the main objective of this study. To achieve this objective, we decided to implement a new ridesharing platform. In this research domain, we mainly targeted professionals. To verify our hypothesis regarding the suitability, we conducted a survey initially with a sample of 150 professionals (65.7% male), and we were able to identify that the majority (82.1%) consider ridesharing as a viable solution to reduce traffic in urban areas.

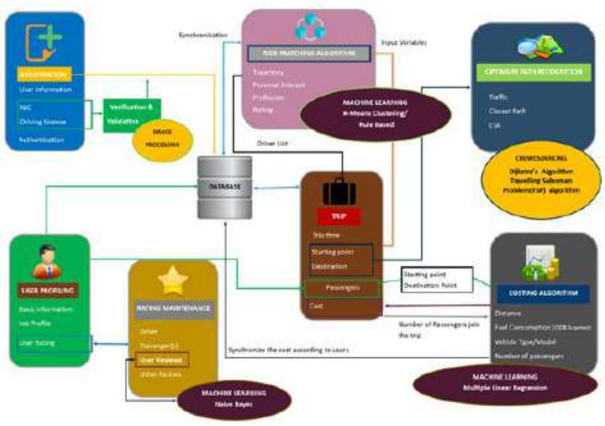

Fig 1: High Level System diagram of the application

After analyzing the existing products and literature, we came up with a set of revolutionized components to create a new ridesharing application(+Go). The main components identified were user profiling management, document validation and rating maintenance, optimum path recognition and dynamic fare calculation which is described in fig 1. Any user who uses our application for the very first time needs to go through the registration process. First, the user needs to verify the mobile number to proceed ahead. After validating the NIC and the license user should provide some basic information such as personal details, preferences, vehicle, and bank details to create the profile successfully. The ride-matching algorithm suggests the most suitable driver list to the passenger upon searching the destination.

In +Go, we always update the database with the current location of all the users using the polling method in every 5 seconds. When suggesting the drivers, it is essential to get accurate information about the location of the drivers as it is one of the main parts of this application to work correctly. Our ride-matching algorithm is divided into two phases; the first phase is used to select the driver list, which matches the preferences, destination, time, starting location and professional level of the passenger. The second phase is used to cluster out those drivers into particular clusters, and the most suitable cluster is retrieved out. For the initial stage; rule-based machine learning is used to select the most suitable drivers for a passenger. In here, we only consider the drivers who in close to 5Km from the location of the passenger. If the driver is beyond the 5Km limit, those drivers are not suggested. When we consider the preferences, we mainly concerned with gender preference, language preference, smoking condition, motion sickness, like quietness and vehicle condition.

In the second phase, the previously selected list of drivers is clustered according to their profession and age. It is essential to highlight that we don't suggest drivers who have a professional-level lower than that of the passenger's professional level. When the drivers are clustered against profession and age; cluster that belongs to a particular passenger is extracted out. Then we'll be checking whether those drivers are being reported from that passenger previously before suggesting them to the passenger. If that particular driver is reported, that person is removed from the proposed list. Finally, all the drivers in who satisfies all the conditions are displayed according to their ratings.

To proceed as a valid user in the phase of registration, the user has to provide valid images of National Identity Card (NIC) and then with the License. In the +Go application, we have given the feature of capturing or loading the images from file, by using the features of android. These images are processed and manipulated in the web server in order to extract the useful information and to ensure the validity. This process is continued as follows.

Three separate algorithms are written to process the images of electronic NIC, non-electronic NIC and License. All these images are first verified for identification of valid human face. This is done using the Haar cascade algorithm. Once it is ensured that valid human face is in the image, the images are subjected to a color changing process where the pixels with dark intensities are turned into red color, and low intensity pixels are turned into white color. This does help in segmenting the useful pixels from the noise in the image.

These color changed images are saved with different names as separate images, and used in the next process. All the image manipulations are done in this component using OpenCV(version 4.1.0) library. Next the above separately saved images are subjected to cropping with the expectation of removing unwanted parts in the images. Then the images are converted to grayscale, subjected to dilation and erosion to remove noises and Gaussian blur with a kernel of 5 is applied to make the image more suitable to be used in text extraction. In the algorithm of processing the image of license, we also have proceeded in changing the brightness, contrast and sharpness of image. Next Tesseract 4.0.0-beta.1 library is used in extracting the NIC number from all the images and also the card expiry date from the image of license. In this text extraction, unwanted characters are filtered out and only the wanted characters are extracted. Also, after extracting the NIC number from the non electronic identity card, the number is converted to the 12 digit format number in order to maintain similarity between the NIC number of electronic identity card. In the authentication process of the application, the two extracted numbers from NIC card and License are equaled for similarity, and if the images are failed in comparison, users are asked to re-upload more precise images.

Rating and user reviews play a vital role in our application in providing a convenient service to the users. +Go application allows the driver to rate the passenger. Also the drivers are given the chance to write a review if it is necessary. Application provides a unique way of rating ability for the passengers. Passengers are given the chance to rate and review the vehicle, the driver and the fellow co-passengers separately. The reviews given are analyzed with the naive bayes classifier to classify and allocate a proper rating. The model is trained with a dataset created by extracting reviews given by users for other popular taxi and travelling applications in the world. For this classification, the stop words in English and some unbiased words found in the dataset are eliminated. Also the single characters in sentiments are removed. By using the naive Bayes algorithm, the probability of words in sentiments are calculated relative to the other words in the sentence, and then the classification rating type with the highest collective probability is identified as the rating of the particular review. The ratings generated through this process is further optimized as we calculate the median of the rating given by the user and the rating generated by +Go system.

Fare calculation plays a vital role in distributing fare among the passengers. There were two types of fare calculations. First one was estimated fare calculation which is predicted before starting the ride, and the other one is the actual fare calculated using the statistics collected, and it will be notified to passengers via a push-up notification at the end of the riding session.

Fuel consumption varies from vehicle to vehicle. So, we surveyed to collect information about the cars which were travelling within the Colombo area. Using the survey we collected, statistics like manufacture year,engine capacity, fuel type, engine power (kW), transmission type, travelled mileage and average fuel consumption. Then we created a multiple linear regression model to predict the fuel consumption according to the vehicle condition.

The number of kilometers that could be travelled using one liter of fuel can be extracted as an output of the multiple linear regression model described above. We identified that it is reasonable to use the following properties for an estimated fare calculation: (i) the price of a liter of fuel in Sri Lanka market(c); (ii) distance(d); (iii) number of passenger/s joined with the ride(p); (iv) fuel consumption(n) in km/l. Equation (1) is derived as a result of the study to calculate the estimated fare of a user.

$$\text{Estimated fare calculation} = \frac{\left(\frac{c}{n}\right) * d}{p} \tag{1}$$

Make	Model	Year_of_Man	Cylinders	Fuel	Capacity	kW	Mileage	Fuel_consumption
SUZUKI	WAGON R	2018	3	P - Hybrid	658	38	981	18
SUZUKI	WAGON R	2018	4	P - Hybrid	658	38	7837	18.3
SUZUKI	WAGON R	2018	3	P - Hybrid	658	38	5337	17.8
Suzuki	Swift	2017	4	P - Hybrid	1242	66	18735	13
MARUTI	ALTO-K10	2014	3	P	998	50	19192	18
MARUTI	ALTO	2015	3	P	796	48	27133	14
MARUTI	ALTO	2015	3	P	796	48	44242	16.3
TOYOTA	PRIUS	2013	4	P - Hybrid	1798	73	54001	12.9

Fig 2: Vehicle Information Data Sheet – Sample

Dijkstra's Algorithm and the Travelling Salesman Problem (TSP) algorithm can identify the closet path among the other routes. Dijkstra's algorithm provides the optimal path to reach the destination by considering the weight of the edges. According to the user's concern, the weight of the graph gets vary. If the user's concern is the distance factor, the length of the edges will be identified as the weight. Similarly, if the user's concerns about traffic, accidents or road closures, weight becomes the calculated time duration to cover the specified distance.

Travelling Salesman Problem (TSP) Algorithm provides the optimal path to travel through a city, which covers all the user-specified locations. When users set their locations, our supposed system will identify the order of the locations to visit. Then this algorithm finds the optimal path, which travels through each location (node) in the listed user-specified locations. Here, registered users will be able to enter the live updates on the relevant path within a specified time range by uploading pictures of the particular incident.

IV. Results and Discussion

The present study was conducted to find an effective solution to reduce traffic congestion in urban areas. The results obtained were put through statistical analysis and are presented in this chapter. For the better understanding the results were divided and presented in a more meaningful manner.

The data from the survey show that the majority of the people wants to collaborate with ride-sharing to reduce traffic congestion recorded with 72.5% emphasizing our main objective. Table 1 shows the percentile breakdown of the reasons stated by the people who opt for ride-sharing.

Table 1: Percentage breakdown of reasons defined by professionals to use ride-sharing

Reason	Percentage Breakdown (%)
To save fuel costs	68.3
To minimize traffic congestion	72.5
To save car ownership costs	28.3
To meet interesting people	17.5
For the benefit of the environment	28.4

A. Matching most suitable driver

The lack of flexibility provided by the existing applications in suggesting suitable drivers to the passengers quoted as one of the primary reasons for the people to hesitate in using ride-sharing platforms[43]. Our dynamic approach has performed particularly well in giving out more optimized results to the expected user. Ride matching algorithm is an important finding in suggesting the most suitable driver list with recorded accuracy of 98%. Despite the accuracy, the number of parameters used to determine the suitable drivers directly proportional to the accuracy of the algorithm produced. In the below diagram, it is clear that, when the input parameters are high, the accuracy of the results produced is too get increases.

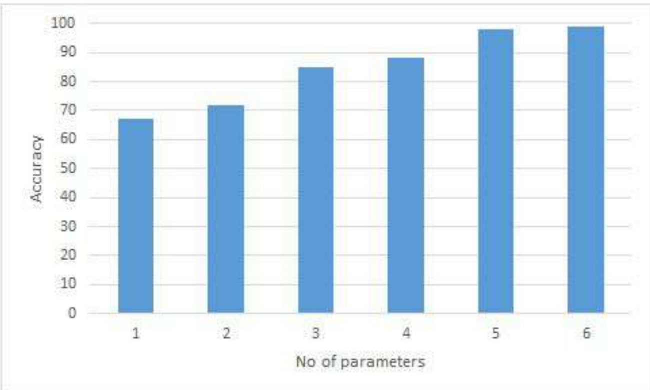

Fig 3: Accuracy vs Parameter count

A similar set of results were obtained by Ghoseiri, Haghani, and Hamedi (2011) matching routes based on the age, smoking condition, gender and occupancy preference. However, the accuracy of the results obtained from the study was not optimized like our study and Jung, Jayakrishnan, and Park (2013) used the same algorithm and provided a solution for the problem associated with optimization. However, in line with the ideas of C. C. Tao and Chen (2008), it can be concluded that this ride-matching algorithm can be more optimized using greedy heuristics.

One concern about the findings was that when the number of parameters increases the complexity of the algorithm gets increases. This causes the algorithm to process slower than usual processing speed (0.86s). Because of this potential limitation, we apply smart data structures rather than moving into sophisticated optimization approaches.

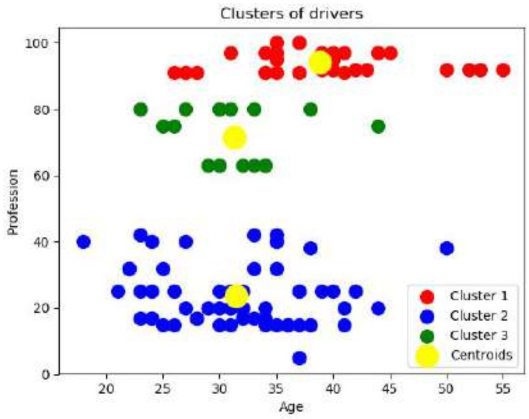

Fig 4: Distribution of drivers across the plane

Aligning with fig. 4(X axis-Age and Y axis-Profession), the selected set of drivers are scattered into several clusters using our ride-matching algorithm. The particular cluster which belongs to the particular passenger is extracted out and displayed.

B. Real time validations

Identification submitted documents are one of the primary sources for verifying the identity of citizens. With the dataset collected from actual images of national identity card and license, both produced more than 83.4% of accuracy in extracting numbers using the image processing done by OpenCV and text extraction by library Tesseract version 4.0.0-beta.1. The results of the experiment found it clear that the Tesseract does decent work in extracting text from the colour images. But it does not always perform well in extracting data from the color images. Hence the accuracy of the process was further increased when the images were subjected to noise reduction from erosion and dilation and then converted into grayscale and next into a binary image. Also, it was observed that the processing time of character extraction from grayscale images is reduced by 10% to 50% compared to colour images. This delivers significantly better results because Tesseract works fast with better text extraction accuracy when it comes to grayscale images and binary images. This result ties with previous studies wherein Chakraborty and Mallik (2013) explain, the more accurate recognition is seen when the color image is converted to grayscale. The table 2 below is a sample test result of text extraction from a low contrast image of a license, which was experimented in the analysis.

Table 2: Text extraction from different types of same image

Image type (Noise removed)	**Property Extracted**	
	NIC number	***Expiry Date***
Raw image	Not Found	Not Found
Grayscale	Found (100%)	Not Found
Binary	Found (91.7%)	Found (100%)

An apparent limitation of the method is that it depended heavily on the position of the image taken for the validation. Higher the background noise, lower the accuracy of the validation. Though we have taken steps to remove noise from the background after the image is uploaded, it's still not possible to remove all the noise in the image uploaded by the user. This introduces a potential confound in user to upload several versions of the image to get it validated from the system.

C. Computation of Dynamic fare

The static pricing methods assign a fixed price for all the users who are on the platform, and it's obviously not fair depending on the distance and time they travel. Results obtained cast a new light on the fare calculation by introducing a new equation to calculate the fare of the ride. Equation (2) calculates for each segment created throughout the ride. Inputs of the equation are the cost for fuel consumption within the segment (f), the number of current passengers (p), and the current price of the fuel in Sri Lanka (price).

$$C_i \text{ (Total fare for the i}^{\text{th}}\text{ segment)} = \frac{(f * price)}{\sum p} \qquad (2)$$

With the fare of one segment is calculated, we add all the segments together to find out the total fare of the ride using(3). Overall these findings are in accordance with the study by Riquelme, Banerjee and Johari (2015), which gives out more accurate fare than the static calculation approach.

$$\text{Total fare of the ride} = \sum_{i = start\ point}^{end\ point} (C_i) \qquad (3)$$

Although this is widely accepted, it suffers from some limitations due to the use of OBD 2 scanner to read fuel consumption using PID commands. As some of the older and different brands doesn't support OBD, it is not possible to start the ride.

In the estimated fare calculation 80% of the dataset is reserved for the training and the rest of the data (20%) reserved for the testing. 93.75% of accuracy is obtained from the Multiple Linear Regression model created to find the estimated fare with the collected dataset. Hence, from the heatmap(Fig. 5), it was clear that several factors affected the fuel consumption including manufactured year, capacity, mileage, engine power and cylinders.

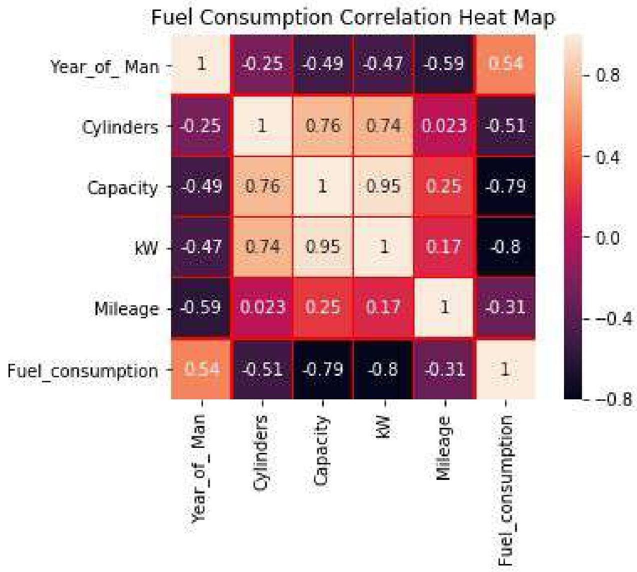

Fig 5: Dependency Correlation between factors

From the study we were able to find four variables that are negatively correlated, and the rest of the variables are positively correlated with fuel consumption given in table 3.

Table 3: Correlation Coefficient of factors

Variable Name	**Correlation coefficient**	**Description**
Manufacture Year	0.54	Moderate positive correlation
Cylinders	-0.51	Moderate negative correlation
Engine Capacity (cc)	-0.79	Strong negative correlation
Engine Power (kW)	-0.8	Strong negative correlation
Mileage	-0.31	Moderate negative correlation

D. Optimum path analysis

Use of Google API for the path analysis is a good solution, but it is not the best when it comes to handling several requests at once. With the introduction of optimum path analysis using the Dijkstra algorithm and Travelling Salesman Problem, we are able to provide the most optimum path considering traffic and distance as parameters handling considerable amount of requests at once. The results confirm that this is a better choice for path analysis with the combination of our crowdsourcing platform.

Table 4: Distance Comparison

Source	**Destination**	**Route Distance(km)**	
		Proposed System	***Google Map***
Malabe	Kollupitiya	16.7	16.7
Dehiwala	Rajagiriya	9.37	9.4
Koswatta	Dematagoda	9.9	9.9
Thalahena	Gothatuwa	7.65	7.7

According to the table 4, proposed system suggest more optimized path at the current moment using crowdsourcing platform and our algorithms. It is not much deviated from the Google API but it gives a less traffic specific path.

E. Sentimental Analysis of ratings

With the collected dataset of reviews from popular taxi service applications in the world; were used for testing and training data. And the selected set of reviews have given more than 85% accuracy in identification of sentiments as positive or negative and 79% accuracy in giving the exact rating with relevant to tested dataset. The results of the study found clear support for the analysis of ratings given by users with consistent results found in previous study done by Baid, Gupta, and Chaplot(2017), where Naïve Bayes algorithm gave the highest accuracy when compared with the other algorithms. In line with previous studies done by Ramya and Rao(2018), more accurate results were obtained by using 80% of the dataset for training and 20% for testing. The approach used suffers from the limitation that the accuracy of the algorithm sometimes get changed when the word density of the dataset does not have considerable amount of matching words to the sentiment provided. In such a situation, the accuracy can be increased by adding similar types of sentiments in the training data set.

V. Conclusion and Future work

Intelligent Complementary Ride-Sharing System, which has been enriched with several machine learning and image processing techniques, has proven that the results are produced with more distinguished accuracy in the context of ride-sharing, and implement a better explanations for its users as a mechanism to diminish the traffic congestion in the country. The principal context of the application; ride matching algorithm has proven with analysis outcomes for its high precision. Moreover, the algorithms such as K-Means Clustering, Rule-based Machine Learning, Naive Bayes Algorithm, Multiple Linear Regression, Dijkstra's Algorithm and Travelling Salesman Problem Algorithm used in the application have made it evident that +Go ride-sharing platform betters other existing applications in the market. Hence, this application will be resulting in reducing the traffic congestion occurs in the urban areas of the country with greater outcomes.

The profile validation of prospective users will be further optimized after the application is launched for the public for a considerable amount of time, and with the feedback or the confirmation of the existing professionals in the application. The usability and the business paradigm of the application will be optimized by introducing a loyalty scheme, and perquisites such as promotion codes to the highly ranked users.

Acknowledgment

+Go research team would like to extend our sincere gratitude to Dr. Darshana Kasthurirathna (Senior Lecturer SLIIT) for consulting us and supporting us to make this venture a success.

References

[1] [online] Available at: http://www.worldometers.info/world-population/sri-lanka-population/ Department of Economic and Social Affairs, Population Division. World Population Prospects: The 2017 Revision. (Medium-fertility variant). [Accessed 10 Jan. 2019].

[2] Vehicleinfo.000webhostapp.com. (2019). *Vehicle Details*. [online] Available at: https://vehicleinfo.000webhostapp.com/

[3] Carpooling.lk. (2019). Carpooling. [online] Available at: http://www.carpooling.lk/ [Accessed 23 Jan. 2019].

[4] Dewan, K.K. & Ahmad, Israr. (2007). Carpooling: A Step to Reduce Congestion (A Case Study of Delhi). Engineering Letters [Accessed 5 Jan. 2019].

[5] R. Manzini and A. Pareschi, "A Decision-Support System for the Car Pooling Problem," *journel on transportation technologies*, Vol.2 No.2 ,2012,pp. 85-101.

[6] Swati. R. Tare, Neha B. Khalate and Ajita A. Mahapadi,"International Journal of Advanced Research in Computer Science and Software Engineering 3(4)", ISSN: 2277 128X April - 2013, pp. 54-57

[7] Fagin, R. and Williams, J. (1983). A Fair Carpool Scheduling Algorithm. IBM Journal of Research and Development, 27(2), pp.133-139.

[8] K. A. Abdul Nazeer, M. P. Sebastian,"Improving the Accuracy and Efficiency of the k-means Clustering Algorithm", Proceedings of the World Congress on Engineering 2009 Vol I, 2009 [Accessed 31 Dec. 2018].

[9] Chih-Ping Wei, Yen-Hsien Lee and Che-Ming Hsu, Empirical Comparison of Fast Clustering. Algorithms for Large Data Sets,2000, Proceedings of the 33rd Hawaii International Conference on System Sciences – 2000.

[10] Ketchen Jr DJ, Shook CL. The Application of Cluster Analysis in Strategic Management Research: An Analysis and Critique. Strategic Management Journal. 1996; p. 441-458.

[11] [online] Available at: https://tradingeconomics.com/sri-lanka/co2-emissions-metric-tons-per-capita-wb-data.html [Accessed 9 Jan. 2019]

[12] K. N. Natei, J. Viradiya and S. Sasikumar, "Extracting Text from Image Document and Displaying Its Related Information", *K.N. Natei Journal of Engineering Research and Application*, vol. 8, pp. 27-33, May 2018.

[13] A. Mordvintsev and K.Abid, OpenCV-Python Tutorials Documentation, Nov 05 2017.

[14] Ding, C.; Wang, D.; Liu, C.; Zhang, Y.; Yang, J. Exploring the influence of built environment on travel mode choice considering the

mediating effects of car ownership and travel distance. Transp. Res. Part A Policy Pract. 2017, 100, 65–80. [CrossRef]
[15] Schneider, R.J. Theory of routine mode choice decisions: An operational framework to increase sustainable transportation. Transp. Policy 2013, 25, 28–137. [CrossRef]
[16] Efthymiou, D.; Antoniou, C.; Waddell, P. Factors affecting the adoption of vehicle sharing systems by young drivers. Transp. Policy 2013, 29, 64–73
[17] Rayle, L.; Dai, D.; Chan, N.; Cervero, R.; Shaheen, S. Just a better taxi? A survey-based comparison of taxis, transit, and ridesourcing services in San Francisco. Transp. Policy 2016, 45, 168–178
[18] R. Valiente, M. T. Sadaike, J. C. Gutiérrez, D. F. Soriano, G. Bressan and W. V. Ruggiero, "A Process for Text Recognition of Generic Identification Documents Over Cloud Computing," 2016.
[19] A. Parwar, A. Goverdhan, A. Gajbhiye, P. Deshbhratar ,R. Zamare and P. Lohe, "Implementation to Extract Text from Different Images by Using Tesseract Algorithm," International Journal Of Engineering And Computer Science ISSN: 2319-7242, vol. 6 Issue 2, Page No. 20298-20300, Feb. 2017.
[20] V. Ohlsson, "Optical Character and Symbol Recognition using Tesseract", Dissertation, 2016.
[21] P. Chakraborty and A. Mallik, "An Open Source Tesseract based Tool for Extracting Text from Images with Application in Braille Translation for the Visually Impaired," International Journal of Computer Applications (0975 – 8887), vol. 68, no.16, April 2013.
[22] S. Ramiah, T. Y. Liong and M. Jayabalan, "Detecting text based image with optical character recognition for English translation and speech using Android," 2015 IEEE Student Conference on Research and Development (SCOReD), Kuala Lumpur, 2015, pp. 272-277.
[23] R. Smith and Google Inc, "An overview of the Tesseract OCR Engine," Proc. 9th IEEE Intl. Conf. on Document Analysis and Recognition (ICDAR), 2007, pp. 629-633.
[24] C. Patel, A. Patel and D. Patel,"Optical Character Recognition by Open Source OCR Tool Tesseract: A Case Study," *International Journal of Computer Applications (0975 – 8887)*, vol. 55, no. 10, October 2012.
[25] L. Dey, S. Chakraborty, A. Biswas, B. Bose and S. Tiwari,"Sentiment Analysis of Review Datasets Using Naïve Bayes' and K-NN Classifier,"*International Journal of Information Engineering and Electronic Business,* vol. 6, Issue 4, pp. 54-62*,* 2016.
[26] P. Baid, A. Gupta and N.Chaplot, "Sentiment Analysis of Movie Reviews using Machine Learning Techniques", 2017.
[27] Llombart, Òscar Romero, "Using Machine Learning Techniques for Sentiment Analysis," 2017.
[28] V. U. Ramya and K. T. Rao,"Sentiment Analysis of Movie Review using Machine Learning Techniques," *International Journal of Engineering & Technology*, vol. 7, pp. 676-681, 2018.
[29] Vaibhavi Patel and Prof. ChitraBaggar, "A Survey Paper of Bellman-Ford Algorithm and Dijkstra Algorithm For Finding Shortest Path in GIS Application," *International Journal of P2P Network Trends Technology(IJPTT*), vol. 5, February 2014
[30] Ni Kai, Zhang Yaoting, Ma Yuepeng, "Shortest Path Analysis Based on Dijkstra's Algorithm in Emergency Response System," *Indonesian Journal of Electrical Engineering and COmputer Science*, vol. 12, no. 5,pp 133-139, 2015
[31] Dechuan Kong ,Yunjuan Liang, Xiaoqin Ma, Lijun Zhang "Improvement and Realization of Dijkstra Algorithm in GIS of Depot"
[32] Ojaswa S. , Darka M. , Francois A. and Girija D., 2005.TRAVELING SALESPERSON APPROXIMATION ALGORITHM FOR REAL ROAD NETWORKS Technical Report 388, Department of Geomatics Engineering, University of Calgary 2500 University Drive NW, Calgary, AB, Canada, T2N 1N4
[33] Adewole Philip, Akinwale Adio Taofiki and Otunbanowo Kehinde, "A Genetic Algorithm for Solving Travelling Salesman Problem," *(IJACSA) International Journal of Advanced Computer Science and Applications*,vol. 2, no.1, January 2011
[34] Dweepna Garg and Saurabh Shah, "Ant Colony Optimization For Solving Traveling Salesman Problem," *International Journal of Computer Science and System Analysis,*vol. 5, no. 1, January-June 2011, pp. 23-29, ISSN 0973-7448
[35] M.I.C., Tunon and M.R. Lopez. "Design and Use of the CPAN Branch and Bound for The Solution of the Travelling Salesman Problem (TSP)," International Conference on Electronics, Communication and Computers, Conielecomp, 1 August 2005.
[36] Pragya, M.Dutta, and Pratyush. "TSP Solution Using Dimensional Ant Colony Optimization," International Conference on Advanced Computing & Communication Technologies, 6 April 2015.
[37] M.M. Mena, R.H. Ucan, V.C. Vetina, and F.M. Ramirez, "Web Services Composition using the Bidirectional Dijkstra Algorithm," IEEE Latin America Transaction, vol. 14, Issue 5, May 2016
[38] A. Ratnasari, F. Ardiani, F. Nurvita. "Penentuan Jarak Terpendek dan Jarak Terpendek Alternatif menggunakan Algoritma Dijkstra serta Estimasi Waktu Tempuh". Universitas Islam Indonesia. ISBN: 979-26-0266-6. Yogyakarta 2013
[39] Zoepf, S., Chen, S., Adu, P. and Pozo, G. (2013). *The Economics of Ride-Hailing: Driver Revenue, Expenses and Taxes*. [ebook] MIT Center for Energy and Environmental Policy Research, p.1. Available at: https://orfe.princeton.edu/~alaink/SmartDrivingCars/PDFs/Zoepf_The%20Economics%20of%20RideHialing_OriginalPdfFeb2018.pdf
[40] O. Santos, D. and C. Xavier, E. (n.d.). *Dynamic Taxi and Ridesharing: A Framework and Heuristics for the Optimization Problem*. Eduardo C. Xavier, p.1.
[41] Riquelme, C., Banerjee, S. and Johari, R. (n.d.). *Pricing in Ride-share Platforms: A Queueing-Theoretic Approach*. [ebook] pp.6,7,15. Available at: http://www.columbia.edu/~ww2040/8100F16/Riquelme-Johari-Banerjee.pdf
[Accessed 6 Jan. 2019].
[42] Aleksandar, P., Silvana, P. and Pancovska Valentina, Z. (2015). *Multiple Linear regression model for predicting bidding price*.
[43] H. Tsao, D.-J. Lin, Spatial and temporal factors in estimating the potential of ride-sharing for demand reduction, Tech. rep., Institute of Transportation Studies, University of California, Berkeley (1999).