2019 International Conference on Advancements in Computing (ICAC)
December 5-6, 2019. Malabe, Sri Lanka

Plus Go: Intelligent Complementary Ride-Sharing System

Viraj Wickramasinghe, Ashane Edirisinghe, Surath Gunawardena, Athrie Gunathilake, Janaka L. Wijekoon, Darshana Kasthurirathna
Faculty of Computing, Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
vjanuradhawick@gmail.com, ashane.edirisinghe@gmail.com, surathruwan@gmail.com, athrienathasha@gmail.com, janaka.w@sliit.lk, darshana.k@sliit.lk

***Abstract*—Currently the world population is gathering to the cities making huge traffic congestion throughout the day. This has drawn serious attention to the society incurred to implement smart solutions for traffic management. One of the prominent problems for traffic congestion is the number of vehicles entering the cities is high. It is a popular fact that the solitary travelers coming to a defined destination make the vehicles underutilized. Therefore, this study proposes a solution to implement a new ride-sharing platform: Plus Go, to reduce this underutilization. Plus Go matches the travelers by considering the designation, traveler preferences, shortest path details, and the ratings of the users. Moreover, Plus Go intelligently estimates the traveling cost based on the fuel consumption of the vehicle, distance traveled, and the time taken to reach the destination. The proposed solution matches the travelers with 98% accuracy ensuring that ride-sharing is an effective solution to reduce the number of vehicles entering the cities.**

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

I. INTRODUCTION

According to the report of the 2017 World Population Prospects of 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 the police reports, there is a gradual increase in vehicles that enter to Colombo during rush hours [2]. 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 feedback received from several frequent traveling staff members within Colombo, we identified that low occupant vehicles like cars that have been in high quantity and resulted in traffic congestion [3]. This phenomenon wastes financial resources and pollutes the environment severely. It has been stated that there is an Rs.500 million loss due to the daily traffic congestion in Sri Lanka [2].

In the initial phase, the proposed system provided only to the working professionals in Sri Lanka. This leads to building a network among professionals. Meanwhile, our proposed ridesharing application helps 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 a possible 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 the proposed method [4].

Furthermore, we surveyed to get the real-time fuel consumption of the vehicles in the urban areas [5]. 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 travel, time, distance, ownership of the vehicle, personal preferences influenced in choosing the individual transportation mode [6, 7]. Certain studies have focused on factors like fuel price, demography and safety measures in selecting the transport mode [8, 9]. With our proposed method, we optimize the solutions provided by the previous studies by suggesting drivers, who match the preferences and professional level of the passenger.

Swati, Neha, and Ajita [6] did research to make the ride-sharing application user-friendly for both the driver and the passenger. From the point of algorithms, Fagin and Williams(1983) and later Ajtai et al. [10] proposed an algorithm to get a portion of users willing to share their cars and from that study, they introduced a new scheduling algorithm. The process of grouping individuals who match each other was first done through using K-Means Clustering. Abdul and Sebastian [11] 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 [12]. In this study, we used the 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. The 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 [13, 14]. Literature has explained that the color images have to be converted to grayscale and then to binary using an adaptive threshold for accurate recognition [15, 16]. Literature has also discussed that OpenCV library, Tesseract OCR and Haar Cascade algorithm does a perfect job in image processing, text recognition and human face recognition respectively [13, 15, 17 18, 19].

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 [20, 21]. 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 [22, 23]. 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 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 [24]. 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 extends to the earliest smallest point.

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 [25]. 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 [26, 27].

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 [28]. Some algorithms have been developed to solve TSP. From the study of Dweepna Garg and Saurabh Shah (2011), they used some other method to solve the Travelling Salesperson Problem, which is known as Ant Colony Optimization [29]. Tunon and Lopez (2005) proposed the algorithm of Branch and Bound to solve Travelling Salesman Problems [30]. 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) [31]. 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 [32, 33].

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 [34]. According to a study by Riquelme, Banerjee, and Johari, they identified two methodologies for pricing as static and dynamic pricing [35]. By scanning throughout the literature, it was obvious that there is no fixed way to calculate the fare for a single passenger. Hence, we used the Multiple Linear Regression method to predict fuel consumption, which results in estimating the fare for the ride.

III. METHODOLOGY

Finding an effective 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, the proposed method decided to implement a new ridesharing platform for the professionals. To verify our hypothesis regarding the suitability, a survey had been conducted 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.

The main components identified were user profiling management, document i.e., driver's license, National ID card (NIC) validation and rating maintenance, optimum path recognition, and dynamic fare calculation. Any user who uses +Go 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 licensed user should provide 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. To identify the current location of users, we update a database using a polling technique at each 5-second interval with their current location.

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. Here, we only consider the drivers who are 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; the 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 who satisfies all the conditions are displayed according to their ratings. To proceed as a valid user in the phase of registration, the user

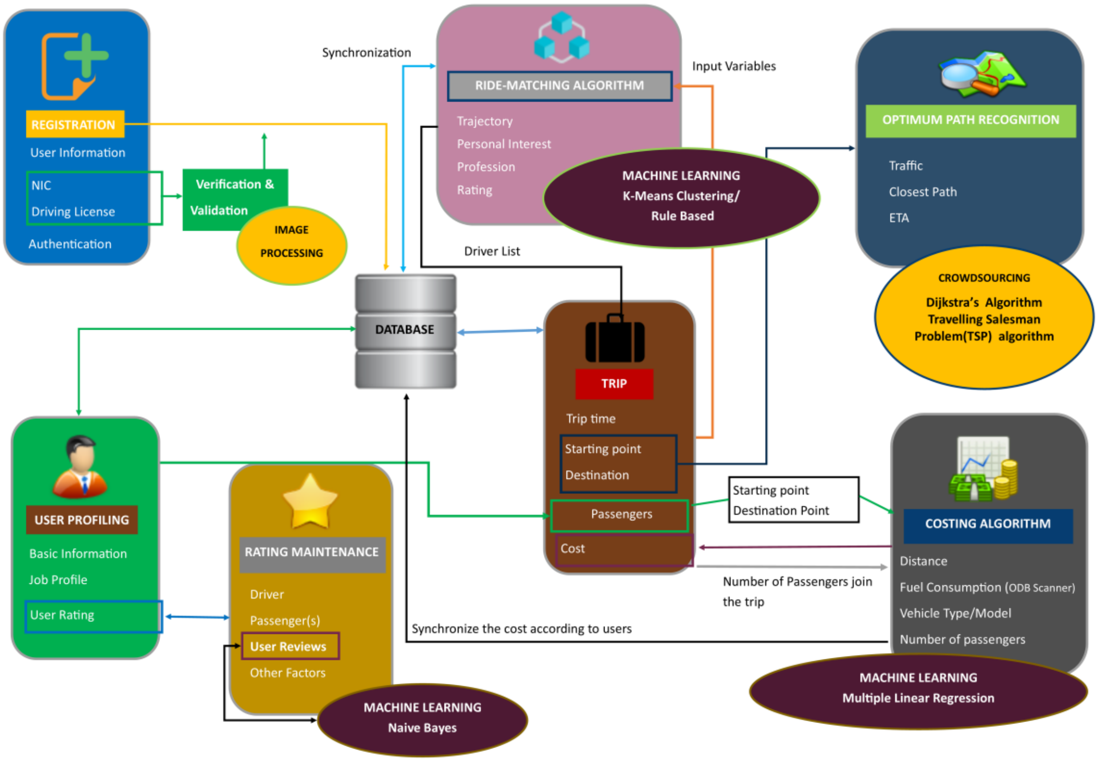

Fig 1: High-Level System Diagram of the application

has to provide valid images of the 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 useful information and to ensure 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 the identification of valid human faces. This is done using the Haar cascade algorithm. Once it is ensured that a 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 the license, we also have proceeded to change the brightness, contrast, and sharpness of the 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 numbers 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.

As Fig.1 continues to describe further about the rating and user reviews which 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. The application provides a unique way of rating ability for 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 traveling 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 is 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 are further optimized as we calculate the median of the rating given by the user and the rating generated by the +Go system.

Fare calculation plays a vital role in distributing fare among the passengers. There were two types of fare calculations. The 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 traveling within the Colombo area. Using the survey we collected, statistics like manufacture year, engine capacity, fuel type, engine power (kW), transmission type, traveled 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 traveled 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.

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

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 a 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 want 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 the most suitable driver*

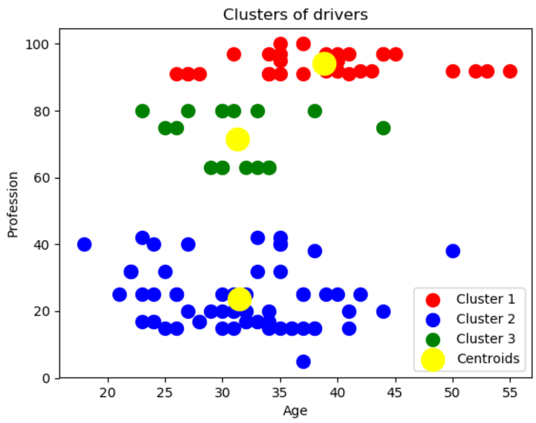

Fig 2: Distribution of drivers across the plane

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]. Ride matching algorithm is an important finding in suggesting the most suitable driver list with an accuracy of 98% giving out more optimized results. A similar set of results were obtained by Ghoseiri et al. [36] matching routes based on 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 Tao et. al [37], 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 increased. 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. Aligning with Fig.2 (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 cards and licenses, both produced median accuracy of 87.5% invalidation process using the image processing and text extraction done by OpenCV and Tesseract(version 4.0.0-beta.1) library respectively. Following fig.3 shows the graph generated based on the text extraction accuracy obtained for each image in the collected dataset.

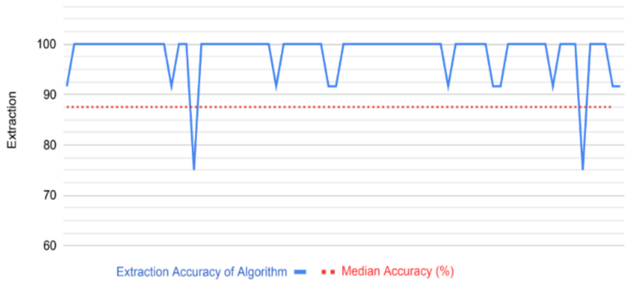

Fig 3: Accuracy percentage analysis of the text extraction algorithm

The results of the experiment found it clear that the Tesseract library does decent work in extracting text from the color 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, then converted into grayscale and next into a binary image. 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. 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 the 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.

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

Although this is widely accepted, it suffers from some limitations due to the use of the OBD 2 scanner to read fuel consumption using PID commands. As some of the older and different brands don'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. 85.08% of accuracy is obtained from the Multiple Linear Regression model created to find the estimated fare with the collected dataset.

Table 2: 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

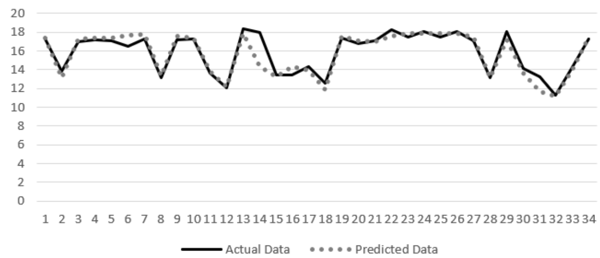

Fig 4: Model accuracy (Actual data vs Predictive data)

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 2.

D. *Optimum path analysis*

The use of Google API for 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 a 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. According to table 3, the proposed system suggests a more optimized path at the current moment using a 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*

The collected dataset of reviews from popular taxi service applications in the world; was used for testing and training data. And the selected set of reviews has given more than 85% accuracy in the identification of sentiments as positive or negative and 79% accuracy in giving the exact rating with relevant to tested dataset. The approach used suffers from the limitation that the accuracy of the algorithm sometimes gets changed when the word density of the dataset does not have a 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.

Table 3: Distance Comparison

Source	Destination	Route Distance(km)	
		Proposed System	*Google Map*
Malabe	Kollupitiya	16.7	16.7
Dehiwala	Rajagiriya	9.37	9.469
Koswatta	Dematagoda	9.9	9.9
Thalahena	Gothatuwa	7.65	7.7

V. CONCLUSION AND FUTURE WORK

This paper proposed an Intelligent Complementary Ride-Sharing System: +Go, that matches the ride sharing based on profession and preferences. The yielded results show that the accuracy of the system in terms of ride-sharing is 98%, which leads to diminish the traffic congestion in the country. Usage of K-Means algorithm for drivers' selection showed 95% accuracy without high computational overhead. Moreover, the ride matching algorithm has proven that it can match the profession and preferences with the highest precision. Furthermore, fuel prediction has been done using multiple linear regression that used for the tariff calculation and the yielded results shows that the +Go system can calculate end-to-end transportation cost in 85.8% accuracy for the shortest paths calculated using Dijkstra algorithm and traveling salesman problem algorithms. The +Go system will be expanded in the future using loyalty scheme, promo codes and user feedback methods during the future implementation of the system.

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] [online] Available at: https://www.newsfirst.lk/2017/03/16/rs-500m-loss-incurreddaily-traffic-congestion-transport-authorities-observe/ [Accessed 30 Dec. 2018].

[3] [Online]. Available: https://indi.ca/2015/10/colombo-vehicle-statistics-2015/.
[Accessed: 02- Jan- 2019].

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

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

[6] 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]

[7] Schneider, R.J. Theory of routine mode choice decisions: An operational framework to increase sustainable transportation. Transp. Policy 2013, 25, 28–137. [CrossRef]

[8] Efthymiou, D.; Antoniou, C.; Waddell, P. Factors affecting the adoption of vehicle sharing systems by young drivers. Transp. Policy 2013, 29, 64–73

[9] 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

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

[11] 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].

[12] 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.

[13] 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.

[14] 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.

[15] V. Ohlsson, "Optical Character and Symbol Recognition using Tesseract", Dissertation, 2016.

[16] 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.

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

[18] 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.

[19] 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.

[20] 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.

[21] P. Baid, A. Gupta and N.Chaplot, "Sentiment Analysis of Movie Reviews using Machine Learning Techniques", 2017.

[22] Llombart, Òscar Romero, "Using Machine Learning Techniques for Sentiment Analysis," 2017

[23] P. Baid, A. Gupta and N.Chaplot, "Sentiment Analysis of Movie Reviews using Machine Learning Techniques", 2017.

[24] 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

[25] Dechuan Kong ,Yunjuan Liang, Xiaoqin Ma, Lijun Zhang "Improvement and Realization of Dijkstra Algorithm in GIS of Depot"

[26] 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

[27] 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

[28] 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

[29] 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

[30] 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.

[31] Pragya, M.Dutta, and Pratyush. "TSP Solution Using Dimensional Ant Colony Optimization," International Conference on Advanced Computing & Communication Technologies, 6 April 2015.

[32] 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

[33] 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

[34] 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

[35] Riquelme, C., Banerjee, S. and Johari, R. (n.d.). Pricing in Ride-share Platforms: A Queueing-Theoretic Approach. [ebook] pp.6,7,15.[Accessed 6 Jan. 2019].

[36] Ghoseiri, K., Szidarovszky, F. and Asgharpour, M. A multi-objective train scheduling model and solution..[Accessed 26 Jan. 2019].

[37] T. Chi-chung and C. Chun-ying, "Dynamic Rideshare Matching Algorithms for the Tax ipooling Service Based on Intelligent Transportation System Technologies," 2007 International Conference on Management Science and Engineering, Harbin, 2007, pp. 399-404.