vRLT: Cloud Based Highly Scalable Connected Vehicle Risk Detection and Life Time Estimation System

Suha Bayraktar and Sezer Gören

Department of Computer Engineering, Yeditepe University,
İnönü Mah. Kayışdağı Cad. 326A
26 Ağustos Yerleşimi 34755 Ataşehir – İstanbul, Turkey

Abstract. In this paper, we propose a cloud-based, scalable architecture for connected vehicle risk detection and life time estimation solution. Our conceptual solution collects real-time data from the vehicle itself by using mobile devices and from vehicle to vehicle (V2V) data generated by the cars in the traffic. OpenXC is the vehicle interface which enables a wide-range of real-time data collection from several points of the vehicle. With the help of a mobile device in the car V2V data can be obtained from the nearby vehicles such as approaching ambulance, motorcycle, sensor on the road etc. In the initial scope, we would to prevent accidents in the traffic. Accidents happen due to many situations in the traffic such as bad road conditions, broken vehicle parts, and poor driving habits. OpenXC and V2V data can be further merged, utilized in learning and prediction by using deep learning algorithms to detect early warnings to prevent accidents. By integrating further data systems such as weather conditions, vehicle service center logbooks and car manufacturers' repositories. Such additional data can be added to make stronger predictions for accidents and can further provide life time estimation of vehicle parts as a further benefit.

Keywords: OpenXC, V2V, Risk Detection, Prevention, Deep Learning.

1 Introduction

Connected cars are becoming fact of near future to enable comfortable, easier and lower risk driving. On the other hand, car manufacturers are trying to improve the quality of the cars from every perspective in order to stay competitive in car manufacturing market. Connectivity in the car moves customer experience and satisfaction barrier to a new level which is mostly influenced and pushed by the social media. By the help of connected car, you can get a lot of advantages like real-time traffic status, weather updates, news, live TV, calls, video chat, e-mail reading. As a side effect of these connected car or infotainment systems, there is a certain risk introduced by using these systems during driving. Car manufacturers shall ensure that these car entertainment systems have low risk usage patterns and do not affect the driving and attention of the drivers. Certain measures are already taken by manufacturers e.g. allowing up to 10 seconds screen usage of an app in the infotainment system.

Are these new connected car and infotainment systems enough or satisfactory for next generation cars? For the moment being many customers might be satisfied with these features both as connected car and in car infotainment systems. In this paper we would like to address a further future oriented need, as we expect that introduction of autonomous vehicles in the traffic will require more intelligence on the road. Soon roads will contain hybrid drivers including the ones without drivers. Therefore we need much more safer and well-managed traffic conditions. As a plan for the near future, we would like to go far beyond existing systems and make our cars more intelligent and reactive systems to simplify our driving experience. This will help us to minimize risks while driving, especially with increasing autonomous cars in the traffic.

2 RELATED WORK

Standard telematics are already developed to understand driver's behavior [1] and driving patters such as acceleration, deceleration, changing from the left to right lanes too often, often zigzag driving, approaching too near to the car in front. With such systems drivers are still responsible of themselves. By using standard telematics systems most of the accidents cannot be prevented, if the driver is not obeying the warnings, alerts he gets based on his driving behavior.

However not every driver on the road has such risky driving patterns. Our aim is to predict the risks caused by risky drivers on the road who are approaching the region of other drivers who are using our proposed solution. In addition to risky drivers, we can add other factors such as bad roads, unexpected traffic jams, bad weather conditions such as thunder storm, heavy fog to increase the safety on the road. By using deep learning feature selection algorithm an enhancing the algorithm with further data, the risk prediction system will take all additional risk factors into account. This can be considered as adding new features in the existing algorithm which have certain effect on the risk.

As the risk in the car is highly affected by bad driving behavior, a face detection system shall be installed in the car to detect the driver. By this way a driver who has a normal driving behavior, but driving a car of risky driver shall be updated in the algorithm calculation dynamically. This is an important point as all of a sudden, the risk factor of a car can be changed dramatically, if a standard, low risk driver starts to use the car. In the meantime the damage caused by the risky driver on the brakes, gear box and etc. will be still carried over on the actual risk, if another driver is using the car. In other words, we need to take every possible effect and changing conditions into account to have a high accurate risk calculation.

Before going into further details of our proposal, let us have a further look in the existing systems and their coverage:

- OpenXC is already in use by many mobile application projects as a vehicle statistics collection system. It is an open system used through the OBD-II [7] (on-board diagnostics) interface which is inserted into car and connected through Bluetooth from a mobile device. A mobile device used in the car shall contain a mobile app which

collects data from the OBD-II interface. With the collected data, it is already shown that many useful apps like:
- fuel consumption
- driving pattern analysis [1]
- carpooling

can be developed. Open XC is only addressing and building apps on usage, driving patterns in the integrated car and does not address road conditions and risks that other vehicles, drivers introduce on the road.

- Vehicle to vehicle (V2V) communication system is currently being developed by the European Consortium car2car communication consortium [8]. The design of the system is in progress. It aims to use radio communications systems in the cars such as Wi-Fi and road installed radio infrastructure [8] to send messages from vehicle to vehicle for important events such as approaching ambulance, motorcycle, roadworks. The main aim is to increase safety on the road. V2V system developed by car2car addresses risks caused by road conditions. However, is not taking driver usage behavior into account. Usage or driver behavior is an important factor which can introduce further risks to the road conditions.

As a way forward, we can clearly claim that both systems are complementing to each other. They can act as a perfect couple to develop a future enhanced risk detection system for vehicles in the traffic.

3 PROPOSED SOLUTION

As we would like to reach a general accident prevention system, we need to understand which factors affect accidents. In general we can say that accidents are caused by many factors. Driver behavior is one of these main factors. Other conditions such as bad weather, unstable road conditions, poor maintained vehicles, spare parts are also playing important role in the accidents. Today for better road safety conditions, one of the most required and improvement goal of countries in the world is to prevent accidents on the road. In general accidents are causing family, health issues, damages on the roads, vehicles. It is clear that they have a significant financial impact to the national economy. If the accidents are prevented and moved to near 0% occurrence, many lives can be saved, significant costs caused by damages can be eliminated.

Our proposed solution can serve as a main contributor to general accident prevention system that can be used by countries in the world to improve traffic safety.

Below in Fig. 1, you can see the illustrated steps of such an accident prevention system which can be formed by our proposed solution steps that we will show in the next stages:

Fig. 1. General Accident Prevention System Steps

- Brake Distance Performance Calculation: is an important step to understand the factors affecting accidents on the road. The break distance can be highly affected by poor brakes in the vehicles. Further spare parts, road conditions, driver behavior can also affect the performance.
- Brake Life Time Calculation: serves as an initial risk parameter for an accident. If we can detect the life time of a brake, then we can send early warning to the driver in the car and drivers in other vehicles. Further life time values of other spare parts can be added.
- Risk Detection Calculation: can be computed by taking brake life time into account. We can enhance the risk calcution with additional parameters such as driver behavior, road conditions, life time values from other parts to increase its accuracy
- Traffic Risk Detection:is a general activity to detect risks in the existing traffic. The risk caused by poor brakes can be considered as a traffic risk. Further parameters can be added improve the its accuracy.
- Accident Prevention: can be considered as a general solution by merging different solutions we propose in this paper into account.

Our main motivations with this solution are to:

- Develop an enhanced risk detection system with the help of additional data sources mentioned in the earlier section and integrate spare parts life time estimator to this.
- Gather many of the single developments in this field in one consolidated solution, add new functionality which does not exist today
- Design a cloud based new architecture which is expandable, scalable and future proof.

As an initial design target, information collected from OpenXC and V2V systems will be stored in Cloud Servers. From these servers all subscribed cars can connect and get real-time information for approaching vehicles, drivers risk effect, weather and road conditions while driving in a certain region.

The data stored in the cloud servers will contain a high percentage of OpenXC generated data. OpenXC is a well-defined system which is mainly developed for Ford vehicles. We also see that further manufacturers are planning to introduce OpenXC in their vehicles. Main advantage of OpenXC is the openness of the development environment. The source code of the whole development is provided in Github [6] for developers for free use. It has also various tools that allow quick development on Android & IOS devices.

With the provided OBD-II [7] based OpenXC dongles, developers can quickly start developing connected car diagnostics for mobile devices and create many values added

application for cars which can be used by diverse business domains. Today we see apps such as watching driver behavior and applying certain algorithms with the target to calculate driver risks for insurance companies.
As a further addition to the existing apps, we offer here to develop an application using OpenXC data which can also act as a life time calculator for spare parts of the vehicles. The life time calculator/estimator will predict life time of a spare part by taking many factors into account such as weather, road conditions and driver behavior.

A good example for such a life time calculation is brakes. Brakes are most often replaced parts in the cars. They usually have the lowest life time in the car. When they approach the end of their life time, the braking distance starts to get longer. This introduces a higher risk in the traffic. When the braking distance get longer on a normal road, this shall be used as a factor to calculate the risk in a driving region.
In general driving behavior, weather, road conditions play a certain role in the life time of the spare part. Therefore car manufacturers shall apply a certain discipline to record all replaced spare parts during service and maintenance transactions such as oil, filters, brake, tires and further spare parts to enable their life time calculation. By this way production quality can be compared with the existing and different spare parts producers when they get faulty and replaced in the service centers.

The effect of the low-quality spare part or the driving pattern to the faulty space part can be calculated and detected by our proposed deep learning algorithm in this paper. The main condition here is to get service and spare parts data collected from car manufacturers and combine them with the date from connected cars.

We name our solution as vRLT which stands for Virtual Risk & Life Time Management. As it can be seen in Fig. 3, the main server of the vRLT will run in a NFVI [4] (Network Function Virtualization Infrastructure) based cloud infrastructure as a virtual network function (VNF) [3] which will allow automatic instantiation, scaling, auto-healing of the solution. As the number of cars connected to the system will increase each day this will require fully automated cloud environment.

V2V solution proposed by car2car is only covered as a future design phase here in this proposal due to its in-progress status. The initial design will mainly concentrate on the OpenXC provided data to further extend existing solutions with life time calculator/estimator. This can be considered as an enhanced risk detection system to the existing practices.

In the initial phase of the solution, vehicles connected to the vRLT can benefit from a limited risk estimation provided by weather, road, spare parts conditions and risky drivers in the driving region.

In the next phases we will further enhance the solution with additional data when V2V projects are deployed and required data is generated by the V2V deployed equipment and applications.

3.1 Proposed Logical Architecture

Fig. 2 below presents the proposed logical architecture. The proposed logical architecture has 5 main data resources:

- Connected cars with OpenXC Interface and Android, IOS Mobile Devices
- Mobile Regional Weather Stations
- Car Service Points which transfer service details of spare parts
- Car Manufacturers main servers for spare parts and spare part related data collection
- V2V servers which transfer data for road conditions and cars on the road

With the help of the flexible cloud architecture demonstrated in Fig. 2, new data source points can be added through the generic web services interface.

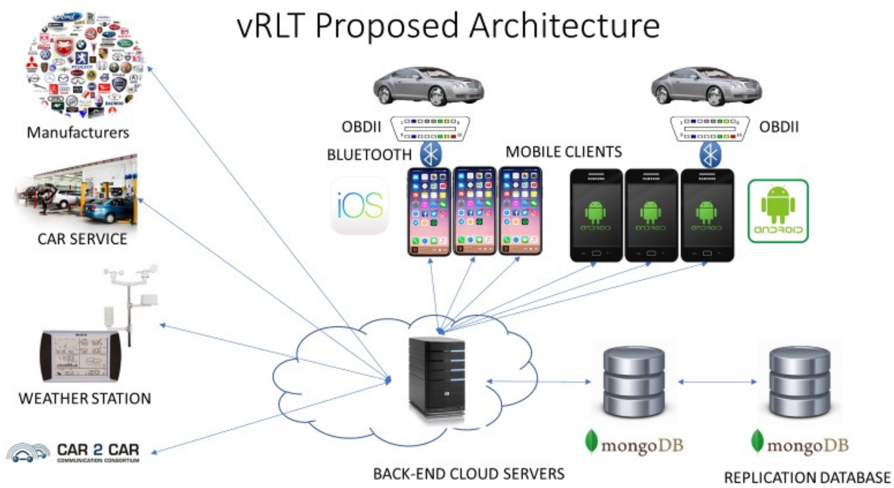

Fig. 2. Proposed vRLT Architecture which shows the relations between data resources, connected mobile clients and storage, and compute servers.

3.2 Proposed Technical Architecture

Below you can see (Fig. 3) Network Function Virtualization Infrastructure (NFVI) [4] architecture which has been developed by ETSI for Virtual Network Functions (VNF) [3] deployments in the cloud infrastructure. VNF's are considered to be next generation cloud applications that bring many flexibilities such as scalability, high availability in cloud-based deployments. Our main solution will be deployed by using this architecture's main components as a baseline architecture.

Base Architecture based on ETSI MANO. MANO stands for Management & Orchestration [3] and acts as a management architecture for multiple virtual network functions that are deployed in cloud. The main advantage of MANO architecture is to provide the ability of auto-initiation, auto-scaling, auto-healing for cloud virtual functions. These auto-enabled features are triggered by monitoring functionality which is the heart of the design that ensures high availability for the running virtual functions. This auto-enable or trigger functions are the part of the closed loop automation that takes automatic actions. These actions are triggered based on the real-time events occur in the application deployment environment. The idea behind using this architecture is to simplify operational environment so that our virtual functions vRLTC, vRLTG, vRLTML scales, improve automatically when the existing environment grows, and unexpected problems occur.

In the ETSI MANO architecture VNF Manager [3] is responsible for providing these close loop functions. By the help of the orchestrator, vRLT Gateway, vRLTML and multiple vRLTC instances can be deployed automatically. The same architecture also allows easy enablement and deployment of new virtual functions in the cloud environment by using VNF Manager and VIM (Virtual Infrastructure Manager) [3].
MANO architecture is built using Openstack [9] virtual environment which will be valid also for our design. Openstack is built as an open source software for creating private and public clouds.

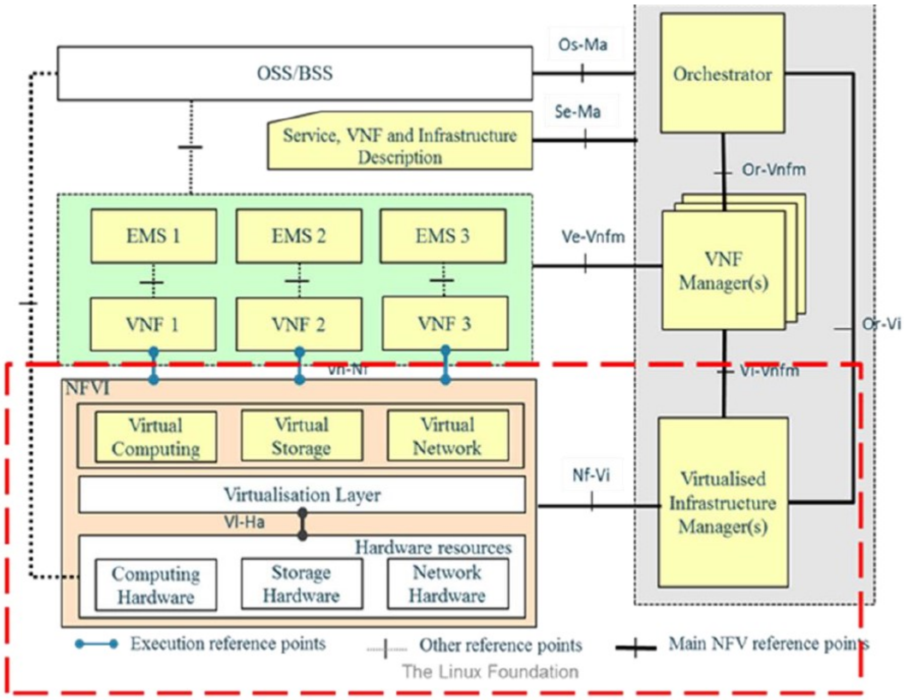

Fig. 3. Standard ETSI MANO [10] Architecture diagram for vRLT modules which will be deployed as VNFs

vRLT Technical Architecture. In Fig. 4 you can see the deployed components of the technical architecture. The connection to the vRLT system from clients will be offered through web services API which is also used by the Android and IOS clients. The proposed vRLT architecture will mainly run with multiple MongoDB instances to collect and restore data on multiple MongoDB Servers. The main aim of multiple MongoDB instances is to provide flexible data storage architecture. The proposed MongoDB architecture allows automatic replications to protect sensible data and allow high availability of the data on multiple replicated data stores.

- vRLTG(Gateway): will be responsible for authentication of connected cars and integrated virtual resources as a gate keeper. These resources will connect to the vRLT for data collection, manipulation and compute. vRLTG will further load balance the connected vehicles to the deployed vRLTC instances so that connections can be distributed equally.
- vRLTC (Car Data): Each instance will manage certain connected devices. Deployed vRLTC instances will connect and register through the vRLT gateway so vRLTCs can provide required functionality to connected clients through web services.
- vRLTML(Machine Learning): modules will be responsible to provide adaptive machine learning algorithms on the collected data for risk and life time calculation. In the design of the vRLTML, we will use a stacked autoencoder [11] to train, learn and extend the feature set impacting the life time of spare parts and further calculate the risk based on life time, driver behavior, weather, road conditions, auto-manufacturer and V2V data. The proof of concept will concentrate on brakes and brake distance calculation.

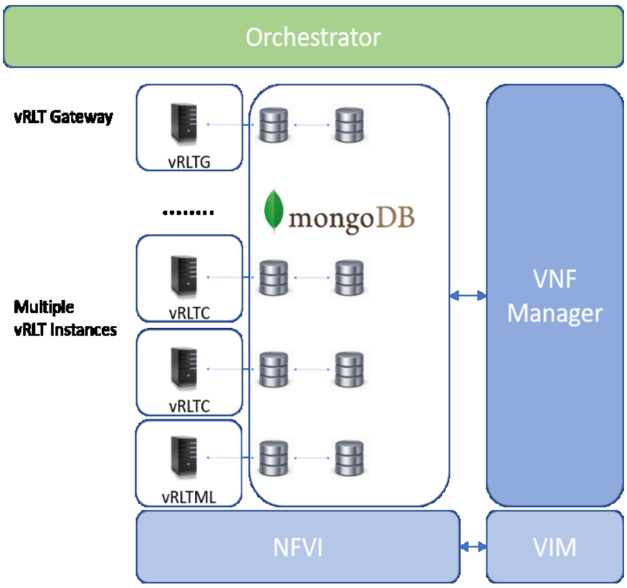

Fig. 4. vRLT Technical Architecture

Brake Distance Calculation Algorithm. From OpenXC [2] provided data, we get the following parameters which have an influence on the distance:

- vehicle_speed: Affects braking distance

- brake_pedal_status: will be recorded continuously for many reasons. For example if the brakes are triggered to often in relation to actual speed, brake distance can be affected.
- Latitude: used to locate car position to make distance calculations and record road conditions for other drivers
- Longitude: used to locate car position to make distance calculations and record road conditions for other drivers
- steering_wheel_angle: will be recorded to understand driver behavior
- odometer: shows the distance
- windshield_wiper_status: will be used to detect rain, if works longer than given default parameter
- time: used to calculate deceleration

Vehicle_speed and odometer can be used directly to calculate the distance by taking brake_pedal_status as a further parameter. With the below simple algorithm we can verify status of the brakes. The formula 1 below can be used as a standard distance calculator equation [5] in the algorithm:

$$d = 0.039(V*V)/A \quad (1)$$

where:

- d = braking distance (m)
- V = speed (km/h)
- A = deceleration (m/s^2)

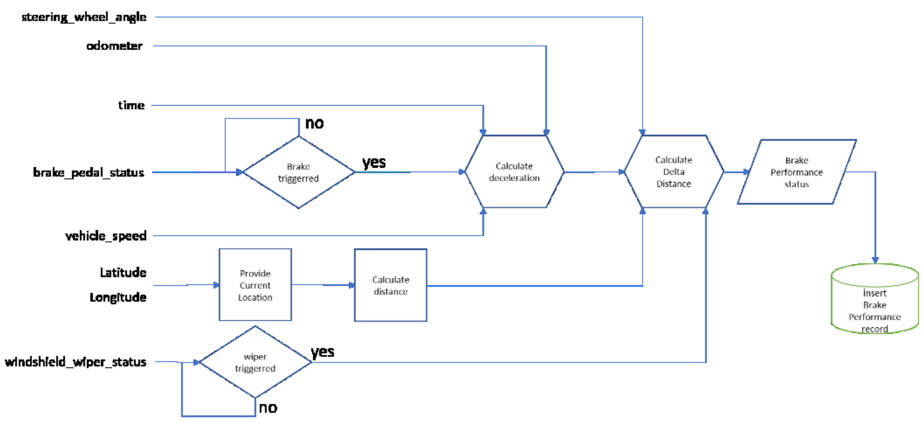

Fig. 5. Brake Distance & Performance Calculation

As a basic result of the algorithm, the life time of the brakes are calculated when the brakes are triggered. If the car does not stop at a calculated distance and there are standard road conditions, it can be concluded that the brakes have already shorter life time. Braking distance is a general study made by many organizations and research projects.

Based on the study made by Trafitec [5] from Denmark, majority of the all motorists brake with a deceleration of more than 4.5 m/s^2, when stopping for an unexpected object on the road. Approximately 90% of all motorist's brake is with a deceleration of more than 3.4 m/s^2. Therefore, when we use 3.4 m/s^2 in our formula we get the following results in the Table 1 based on report from Trafitec:

Table 1. Braking distances based on Eq. 1 [5]

Speed (km/h)	Braking Distance(m)
20	5
30	10
40	18
50	29
60	41
70	56
80	73
90	93
100	115
110	139
120	165
130	194

This allows us further to calculate deviation of multiple brakes from different time frames for connected cars and conclude a life time estimation. This can be finalized and confirmed at a service station when the brakes are replaced with new brake pairs. For further detailed calculation, we need to see the effect of other parameters such as:

- steering_wheel_angle
- anti-lock braking system (ABS)
- road conditions, such as curved, with potholes and bumps and uneven pavements (using latitude and longitude)
- asphalt type
- weather condition
- driver behavior
- total distance
- tire type

These parameters have certain influence on the braking distance including driving behavior which might cause further deviation from standard calculations.

By using the data provided from connected cars, stacked autoencoder based deep learning algorithm can be trained to teach the network target outputs. Brake distance performance we calculate:

Brake Distance Performance = (Measured distance/Calculated distance) *100 (2)

Let us assume that we have brake distance performance ranges of 0-70%, 71-80%, 81-90%, 91-100% as outputs of the autoencoder softmax regression classifier algorithm shown in Fig. 6. After the training, we classify our outputs 81-90% and 91-100% as out of high-risk area.71-80% medium risk and 0-70% high risk.

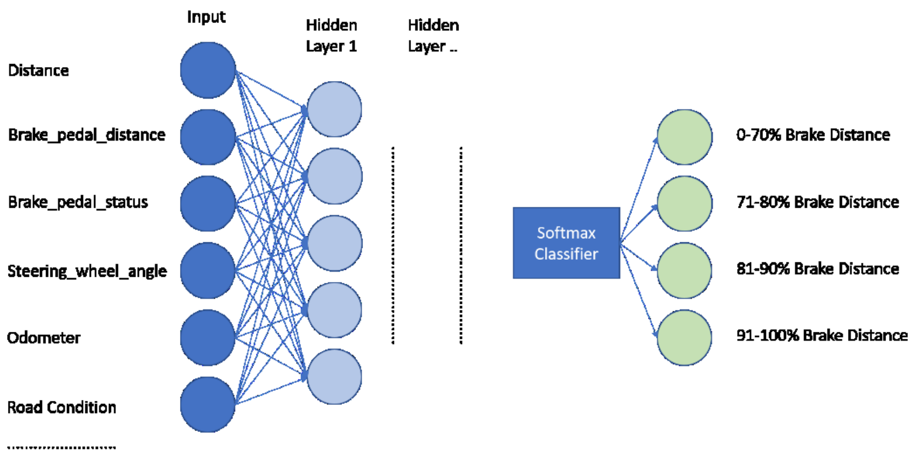

Fig. 6. Stacked Autoencoder Deep Learning Diagram

Above stacked autoencoder algorithm will be used for feature extraction and brake distance outputs. The brake distance outputs will provide a record for the performance of the brakes by using different input parameters given in the diagram.

Spare Part Life Time Estimator/Calculator Algorithm. Spare part life time calculator algorithm is also based on stacked autoencoder algorithm which is shown in Fig.6. This will allow us to train the neural network using same parameters used for brakes with further additional parameters. The main difference here will be simple output which will be based on life time. We can use 10 outputs at the softmax classifier from 1 to 10 years for longer life time spare parts. For the training of the outputs, we will use received life time value of replaced spare parts from service stations. For many of the spare parts such us brakes, we will need 1-4 outputs. As we plan to add further calculation of other spare parts, we will keep 10 outputs due to possible longer life time than frequent replaced spare parts such as brakes.

The life time values received from softmax classifier will be used further in the risk calculation algorithm as a parameter. This parameter will be shared between connected OpenXC vehicles through the vRLT system.

Brake life time estimator outputs will be mapped to life time values based on the following categories:

- 0-70% Brake Life Time: Replacement required, very high risk
- 70-80% Brake Life Time: x months left for replacement, high risk

- 80-90% Brake Life Time: y months left for replacement, intermediate risk
- 90-100% Brake Life Time: z months left for replacement, low risk

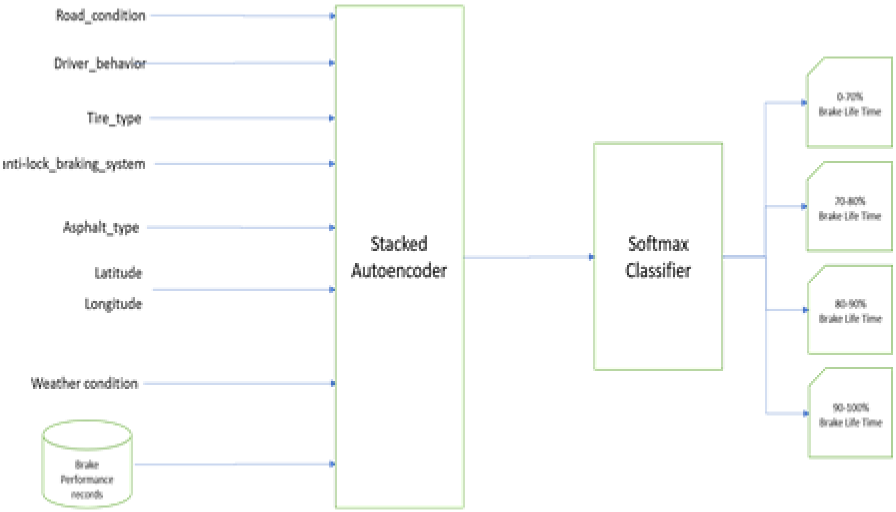

Fig. 7. Brake Life Time Calculator Deep Learning Algorithm Diagram

Above diagram illustrates the brake life time calculation logic using stacked autoencoder deep learning algorithm. The brake performance records are already collected as a result of the initial algorithm shown in Fig. 5. These are merged with additional parameters to detect the effect of new parameters to the brake life time which can be considered as risk detection system. The success of the final algorithm is highly dependent on new parameters added to the algorithm in Fig. 7 at later stages. By this way the final result accuracy can be increased. In this algorithm we did not add the service station spare part replacement data which is a final confirmation for the accuracy of the life time results.

4 FUTURE WORK

Further studies of our proposed solution depend highly on the collected data from vehicles that are equipped with our solution.

4.1 Further Analytics

As an add on the brake system analytics and risk detection system, following analytics are planned in the future as the collected data will increase in time:

Driver Behavior Contribution: Driver behavior plays an important role for the life time of the brakes and further parts such as disc brakes, tires, gears. We also plan to identify effect of different drivers using the same car to the spare parts. As a result of this study automatic warning system can be improved to warn out of bound driver behaviors.

Transmission Type: An automatic transmission and manual transmission are expected to have a different effect to the mentioned spare parts. As a result of the future study, it is assumed that automatic transmission can introduce more life time which we plan to verify in the future.

Automatic Warning Systems for Safety: Many vehicle manufacturers are already equipped vehicles with automatic safety systems with sensors that detect tired drivers, zigzag driving, speeding, distraction by mobile phones and infotainment systems. As a result of this work, further warning system algorithms will be developed to increase car safety.

Vehicle Infotainment Systems & Usage of Mobile Phones: Infotainment systems are already in place in many vehicle models. Drivers use these systems to get entertainment and location navigation. There are various studies made on this topic which is categorized under distracted driving. One of these studies conducted as a Thesis at University of Kansas by Ashleigh V. Tran [12]. The study shows different results based on usage of mobile phone under different road conditions by also taking usage time into consideration. Our future study will also include detecting algorithm that watch the usage of these systems by taking usage time, road conditions into account. This will allow us to create automatic warning systems to decrease accidents and risk situations.

Accident Analytics: It is important to understand how the accident is affected by different driver behavior such as zigzag driving, speeding, distracted driving, road conditions etc. These results will also contribute to detect the effect to the risk we provide in this study.

4.2 Improvement of The Proposed Software Architecture:

One of the key activities in future work will be to evaluate the existing auto scalable, high available architecture. NFVI based MANO architecture developed by ETSI is proved to scale and provide high availability in many SDN/NFV projects in the field. However, it is possible that by increasing data collected from the vehicles, the modules vRLTG, vRLTC and vRLTML need to be watched to detect the effects of increasing data from edge devices. Especially in the area of IoT (Internet of Things), edge devices are further equipped with machine learning analytics after certain experience in the field. The main purpose for those edge computing algorithms is:

- Simplifying back-end processing
- Faster reacting to risk situations

A study is already made by Farzad Samie [13] that shows the ineffective usage of back-end servers for large IoT projects. Especially faster reaction time through edge devices is an important factor that might prevent longer travel time of data from edge device to the back-end servers. The longer reaction of algorithms to the edge device might increase the risk factor in many cases. This will be one of our key studies for the future work.

5 CONCLUSION

In this paper we have conceptually designed a basic risk estimation system which can act as an initial design for future oriented traffic risk detection and safety improvement system. By providing the outputs of the algorithm in Fig. 7, drivers in other vehicles can see high risk vehicles on their destination. It is highly important that by using a deep learning stacked autoencoders algorithm, we allow future extensions to the existing data model to increase the accuracy and success of the solution. This will allow calculation of future risks introduced by new parameters especially received from further V2V data.

Following enhancements can be considered as future enhancements to the initial proof of concept:

- New data resources
- Zone based risk calculation
- Radio Signals on the road

These new resources and algorithms can further allow fine-tuning of existing algorithms and increase confidence level in the general accident prevention system formed by traffic risk detection and spare part life time estimator.

References

1. Sujata Khedkar et al, International Journal of Computer Science and Information Technologies, Vol. 6 (3), 2015, 2738-2745
2. OpenXC platform webpage [online] available: http://openxcplatform.com, 2011.
3. ETSI GS NFV-SEC 013 V3.1.1 (2017-02), network functions virtualisation (NFV) release 3; security; security management and monitoring specification, [online] available: http://www.etsi.org/deliver/etsi_gs/NFV-SEC/001_099/013/03.01.01_60/gs_NFV-SEC013v030101p.pdf
4. Network Functions Virtualisation, [online] available: http://www.etsi.org/technologies-clusters/technologies/nfv
5. Poul Greibe, scion-DTU [unpublished] Trafitec July 2007, braking distance, friction and behavior, [online] available: www.trafitec.dk
6. OpenXC source code, [online] available: https://github.com/openxc/openxcplatform.com
7. Onboard Diagnostic System 2, OBD-II, http://www.obdii.com/
8. Car2car consortium for V2V project, [online] available: https://www.car-2-car.org/index.php
9. Openstack, [online] available: www.openstack.org
10. Network Functions Virtualisation (NFV); Management and Orchestration, [online] available: http://www.etsi.org/deliver/etsi_gs/NFV-MAN/001_099/001/01.01.01_60/gs_NFV-MAN001v010101p.pdf
11. Stacked Autoencoders, [online] available: http://ufldl.stanford.edu/wiki/index.php/Stacked_Autoencoders

12. Ashleigh V. Tran, Understanding Willingness to Respond to Messages and its Relationship to Driving Performance, University of Kansas, [online] available: https://kuscholarworks.ku.edu/bitstream/handle/1808/25992/Tran_ku_0099M_15451_DATA_1.pdf?sequence=1
13. Farzad Samie, Resource Management for Edge Computing in Internet of Things (IoT), University of Karlsruhe, [online] available: https://publikationen.bibliothek.kit.edu/1000081031/7142406