Study on Knowledge Representation Framework and Anomaly Detection of the Intelligent Vehicle

Juan Wang, Haoxi Zhang, Zuli Wang, Jun Zhao and Lei Shi
School of Cybersecurity, Chengdu University of Information Technology, Chengdu, China
Email: wangjuan@cuit.edu.cn

Abstract. Intelligent vehicle is a complex system includes at least three parts of information, the vehicles themselves, driving surroundings and drivers which has different formats, different properties, and different collection methods. These differences bring difficulties to communication and knowledge sharing among vehicles and drivers, and also create barriers to vehicle safety research. To address these problems, we propose a novel intelligent vehicle knowledge framework to contain all of related information of intelligent vehicle with clear hierarchical architecture and unified SOEKS structure. Based on the framework the anomaly detection approach using past driving experience and the neural knowledge DNA (NK-DNA) is proposed. And then we introduce the Long Short-Term Memory (LSTM) neural network into the Networks of NK-DNA, to detect abnormal CAN packages. We examine our approach with real car data and simulated attack data. The simulation results shown our framework and detection method can share easily and found anomalies of CAN bus flow.

1. Introduction
With the rise of Tesla vehicles, the upsurge of informationization and intellectualization of the vehicle industry has been set off. Many artificial intelligence companies have entered the field of intelligent vehicles, such as: the CarPlay of Apple Inc, Android Auto, the Unmanned Vehicle project of Google and Baidu Corporation. However, the intelligent vehicles are becoming new targets for hackers. Almost all intelligent vehicles have entertainment information systems which provide music, video, Internet and other entertainment functions. Hackers can raise attack remotely through WIFI module or 3G/4G SIM card of the entertainment information systems.

Those attacks can make vehicle out of control, and endanger the lives of drivers and passengers. For instance, K. Koscher et al made some vehicular information system intrusion, such as shutting down the brake system or locking passengers in the vehicle [1]. Miller C, Valasek remotely invaded a Jeep Cherokee 100 kilometres away, modifying ECU firmware, changing vehicle speed, shutting down engines, controlling windows, etc [2].

Generally speaking, for intelligent vehicle security the government and industry alliance pay more attention to the security framework design of vehicle, while the research institutes and universities pay more attention to the encryption scheme, security authentication scheme and signature algorithm of vehicle communication. The OVERSEE project of the European Union designed an open automotive platform, which provides a trusted access mechanism [3]. The EVITA project, also belongs to the European Union, carried vehicle safety components to prevent sensitive data from being tampered [4]. The BMW company associated other famous automobile enterprises put forward a set of software architecture, called AUTOSAR, to ensure the safety update of ECU software which contains memory manage, on-board network and diagnostic functions [5]. Schweppe et al. proposed EPSB, Libra-CAN
and other light communication authentication protocols for ECU network of vehicles, and verified
their availability [6-7]. Woo et al. designed a lightweight encrypted communication method using 32-
bit AES algorithm, which greatly reduced the load of vehicle network bus [8]. Weimerskirch and Paar
introduce PKI mechanism to key management and encrypt CAN bus communication [9]. Liu Hui et al.
designed a group management scheme for communication in Vehicular Ad Hoc Network (VANET)
using group signature algorithm [10]. Zhu Xiaoling et al. presented an encryption and decryption
model for vehicle black box using elliptic curve cryptography and secret sharing algorithm [11].
Zhang Wenbo applied proxy signature and threshold signature algorithm to construct trust chain of
TPCM trusted platform [12]. Wang Jingxin et al. designed an anonymous exchange information
scheme using group key mechanisms which can ensure connectivity channel [13].

All the above studies focus on certain issues of intelligent vehicle safety. However, intelligent
vehicle is a complex system. Its security is not only related to the communication inside vehicle or
between vehicles, but also greatly affected by drivers and driving environment. And the data of
vehicles, drivers and driving environment have different formats, different properties, different
collection methods. These problems make it is difficult to communication and knowledge share among
vehicle, driver and driving environment. Similarly, it is difficult to get comprehensive analysis results
from these data. In this paper, a knowledge representation framework of intelligent vehicle is proposed
which offer uniform format for all kinds of intelligent vehicle data and constructed a modularization
and hierarchical information architecture for these data. This framework can provide basis for
comprehensive decision-making of intelligent vehicle safety, such as anomaly detection.

2. Knowledge Representation of Intelligent Vehicle Based on SOEKS Structure
The Set of Experience Knowledge Structure (SOEKS) is a structure of knowledge representation,
which can unify many heterogeneous information into a standard structure for easy combination
and decomposition. In this section, in order to deal with intelligent vehicle data which are multiple sources,
complex formats and different meanings, we redefined the meaning of SOEKS components. The
SOEKS is the combination of four components that characterize decision making actions (see Fig.1) [14]:

Variables (represented as $V_i$, $V_2$,..,$V_n$) are the source of the other components, and use an
attribute-value language (i.e. by a vector of variables and values) to represent knowledge; In intelligent
vehicle area, variables represent meta-data of intelligent vehicle. For a particular level, meta-data
describe the component of special system which cannot be further divided. Vehicle meta-data such as
driver's license ID, driver's gender, vehicle brand, speed, right front door is opening or not are can be
abstracted as variables. The value of a variable is the value of a specific object. For example: speed is
a variable, and 80 km/h is its value.

Constraints (represented as $C_1$, $C_2$,..,$C_m$) are the limitation of variables; For example, the diver's
age is limited to [18-70], unit: year and the speed is limited to [0-450], unit: km/h.
Functions (represented as $F_1$, $F_2$,..,$F_i$) describe associations between a dependent variable and a set of
input variables; For example, the function of "acceleration" acts on the variable "speed" to change the

\[ F(x) = \frac{d}{dt} \text{speed} \]

Figure 1. Set of Experience Knowledge Structure (SOEKS)
value of the variable. In practice, multiple functions may act on different variables at the same time, resulting in changes in the values of multiple variables.

States (represented as \(S_1, S_2, \ldots, S_i\)) which we add into SOESK are used to describe certain state of a vehicle. The state is a set of values of variables. For example: State 1 which represented as \(S_1\), and several values of variables constitute this state as: \(S_1\) = (speed: 100 km/h, mileage: 20003 km, engine water temperature: 100 °C, steering wheel torque: 230), after \(F_1\) = "deceleration" and \(F_2\) = "turning", the vehicle state becomes as \(S_2\) = (speed: 70 km/h, mileage: 20004 km, engine water temperature: 98 degrees, steering wheel torque: 330).

Rules (represented as \(R_1, R_2, \ldots, R_j\)) are series of S and F with temporal sequences. For example: \((S_1, F_1, F_2, S_2)\) can be regarded as a rule. If under the state \(S_1\), affected by \(F_1\) and \(F_2\), the state change to \(S_2\). If the state deviates too much from \(S_2\), it may be abnormal. In a word, rules are suitable for representing inferences, or for associating actions with conditions. They are relationships between a condition and a consequence connected by the statements IF-THEN-ELSE. Using IF-THEN-ELSE statements the rule \((S_1, F_1, F_2, S_2)\) can be described as: (IF \(S_1\), \(F_1\), \(F_2\) THEN \(S_2\) ELSE alert;). The set of all rules is called the set of experiences about intelligent vehicles.

3. Intelligent Vehicle Knowledge Representation Framework Based on SOEKS

This section we design an Intelligent Vehicle Knowledge Representation Framework (IVKRF) to contain all of relation information of intelligent vehicle with clear hierarchical architecture and unified SOEKS structure. "Intelligent Vehicle" involves three main parts related to driving safety: 1. Vehicles themselves, 2. Driving surroundings, 3. Drivers. The three parts constitute the "Top Level" of intelligent vehicle knowledge representation framework. And of course, each component can be divided into lower level compositions which give more details, as the Fig.2 shows.

![Figure 2. The Intelligent Vehicle Knowledge Framework (IVKRF)](image_url)

The whole framework is divided into five layers, and from top to bottom they are: Top, Components, Information, Subsystems and Sensors.

As mentioned above, the Top Level has Driver, Vehicle and Surroundings three parts. And each part has two or three components. For Driver and Vehicle parts, they have the static information
components named Basic Information (Basic Info.) which is independent of the vehicle and will not change. Other components are the dynamic information components named Drive Info. of Driver and System Info. of Vehicle. Drive Info. has close relationship with driver and keep the same for a while except mileage-related parameters. And the System Info. of vehicle contains the dynamic data of each system of moving vehicle. Finally, the Surroundings contains other information related to vehicle driving.

Driver’s behaviour has an important influence on the safety of intelligent vehicles. The driver component has two parts: the basic information of the driver, such as driver’s name, height, weight, etc. This information is static that does not change in a short time and has less relation to driving. Correspondingly, information that has strong relation to driving is classified as driving information, includes: Date of License Issued, License Type, Driving Kilometres, if ever had a driving accident or not, and so on.

Surroundings also greatly affects driving behaviour and may lead to security problem. The first part of surroundings is weather, it is obviously that driving behaviour is different on rainy day and sunny day, and driving on snowy day maybe dangerous. Similarly, the road conditions lead to different driving behaviour too. The road conditions include: Road Type (urban road or mountain road), Road Class (interstate highway or local road), and Traffic (jam or not). The third part of surroundings is Scenario of the driver. When a driver hurried to send the patient to the hospital, her driving behaviour must be different from usual.

The last and the most important component is Vehicle. The static and less related to driving information is described as Basic Information of vehicle, includes: the colour of the vehicle, the size of the vehicle, the manufacturer of the vehicle and so on. The vehicle System Information directly related to driving is divided into four sub-systems: The Power and Transmission system, the Chassis Safety System, the Entertainment Information System and the Vehicle Control System. Each system can be divided into more specific components. Take Vehicle Control System as an example. This system consists of Door, Seats, and Body Control Units. At last, each subsystem can get its data from sensors which are the lowest level.

Using the hierarchical framework and SOEKS we can describe the whole knowledge of the intelligent vehicle well. Here is an example, we use SOEKS describe the vehicle information from the bottom to the top. Firstly, the Sensor layer, the four doors are close or not can be described as: Driver’s left door sensor SOEKS (denoted as DLS):

\[
\text{Sensor}\begin{cases}
\text{<Sensor>} & \\
\text{name} & \text{DLS} \\
\text{variables} & \text{DLS} \\
\text{constraints} & \{1,0\} \\
\text{functions} & \text{NULL} \\
\text{rules} & \text{IF DLS}=0 \text{ THEN return } 0 \text{ else return } 1
\end{cases}
\]

Label <Sensor> represents the sensors layer, and the label <name> used to differentiate SOEKSs. The variable of this SOEKS is DLS. If DLS is equal to 0 means the door is closed, and equal to 1 means the door is open. So, the constrains of DLS are \{0,1\} and the rules of this SOEKS is “IF DLS=0 THEN return 0 else return 1” and the “0” represents safe and the “1” represents dangerous. For this SOEKS, the function is NULL. Other three doors’ SOEKS are similar to this one. So, for sensors layer, we get four doors’ sensor SOEKS: Driver’s left door Sensor(DLS), Driver’s right door Sensor(DRS), Passengers right door Sensor(PRS), Passengers left door Sensor(PLS).

The lower layer SOEKS can be taken as an entirety to be the upper layer’s variable. Follow the example above, we can give the SOEKS of door control system of subsystems layer:

\[
\text{Subsystem}\begin{cases}
\text{<Subsystem>} & \\
\text{name} & \text{Door control systems} \\
\text{variables} & \text{DLS, DRS, PLS, PRS} \\
\text{constraints} & \{1,0\} \\
\text{functions} & \text{F=DLS && DRS && PLS && PRS} \\
\text{rules} & \text{IF F\#1 THEN Alert}
\end{cases}
\]

4
Label $<$ Subsystem$>$ represents the Subsystem layer and other labels’ meanings are similar to above expect functions and rules. The function of this SOEKS is “$F=DLS \&\& DRS \&\& PLS \&\& PRS$” which checks if all doors are closed. And the rule of this SOEKS is “IF $F \neq 1$ THEN Alert” which meanings if there is more than one door is not closed then the car is unsafe, an alert will be rise.

Then the subsystem SOEKS can be taken as an entirety to be the Information layer’s variable, and the Information SOEKS will be variable of Components layer, and so on. Finally, the “Driver SOEKS”, “Vehicle SOEKS” and “Surroundings SOEKS” constitute the whole knowledge of intelligent vehicles. According to different requirements, the above modules and levels can be flexibly combined.

4. Complex Knowledge Representation Based on NK-DNA

Above knowledge framework is sufficient for representation and storage of many experiences of intelligent vehicles. However, complicated experiences need complicated structure to represent and store, like neural network. The model called Neural Knowledge DNA (NK-DNA) [15-18] is combine both SOEKS and neural network as Fig.3 shows. Simple experiences are represented and stored by SOEKS and complicated experiences that need many SOEKS information work together are represented and stored by neural networks.

The core of the model is networks in which store details of neural networks used for training, such as the network structure, weights, bias [19]. Actions are used to store the domain’s decisions set; and the experiences, which are described by SOEKS, contain domain’s historical operation segments with feedbacks from the outcomes. The last component—the States are situations in which a decision can be performed. And for different application scenarios, different structures of neural networks with will be trained and stored in Networks.

![Figure 3. Conceptual Structure of the NK-DNA](image)

In this paper, we focus on the security of intelligent vehicle. According to the internal structure of intelligent vehicles, external attacks have to manipulate relevant components through CAN(Controller Area Network, CAN) bus. It is to say the attacks will add attack CAN packages into normal CAN package flow. Normal driving behaviour corresponds to regular CAN command sequences and attack CAN packages will destroy the sequence characteristics of these normal commands. Based on this principle, we introduce the Long Short-Term Memory (LSTM) structure neural network into Networks to detect abnormal CAN packages. The cell structure as the Fig.4 shows, t is time, and the t-1 is the last time, and the t+1 is the next time. “CAN” is the sequence of CAN commands, not only
one CAN command. The “H” is hidden layer intermediate result. A is a LSTM cell as references [19]. The input is a sequence of CAN commands, and the output is a sequence of CAN commands at the next moment. The step of the CAN command sequence, namely the length of CAN command sequence is set by the experimental results.

5. Simulations and Analysis

5.1. Simulation Data
It is dangerous to carry out attack experiments on real vehicles, so the experimental data in this paper are normal driving data collected by the real vehicle and merge simulated attack data. The real car data are the 500-minute 2010 Toyota Prius driving data provided in reference [20], which is about 120,000 pieces as Fig.5 shows.

![Figure 4. LSTM Cell of CAN Bus Command Sequence](image)

![Figure 5. Driving CAN Packages of 2010 Toyota Prius](image)

CAN data package includes two parts: ID and data. The ID is a priority field, the smaller figure, the higher priority. IDH is the high byte of ID, and IDL is the low byte of ID. Len is the abbreviation of length which means the size of following data. Finally, Data are the content of CAN command. In this paper, we only check ID sequence ignore the data part. The data set merge some attack data with attack label. Analog attack data are generated by Pycharm tool according to CAN protocol format. 80% of the data were used in the training process, and the remaining 20% were used to check the accuracy.

5.2. Inheritance and Sharing of Knowledge Based on SOEKS
Experience presented as SOEKS structure is easy to inherit and share. Take the 2010 Toyota Prius as an example. From its CAN bus data, we found driving operation has very strong frequency regularity, such as the servo steering operation is about 16-18 times per minute, if exceeds this limitation there maybe abnormal. This experience is presented as following SOEKS structures:
<sensor>
  <name>steering</name>
  <variables>CAN ID=266</variables>
  <constraints>{left max angle, right max angle}</constraints>
  <functions>F=\text{sum (CAN ID)}/\text{minute}</functions>
  <rules>if F \in (16-18) then pass else alert</rules>
</sensor>

In fact, the whole Toyota Prius series have similar internal architecture, so above experience can transplant to another version of the same series without change. Other experience can be found in reference [21].

5.3. Anomaly Detection of CAN Command Sequence Based on NK-DNA

First of all, it should be emphasized that the IDs that makes up the sequence have to be strong correlation, if not the temporal characteristics of the sequence can be overwhelmed by lots of noise data. Here we take the sequence which made up by two IDs the 0283 and 0266 as an example sequence. The ID 0283 is braking command and the ID 0266 is steering command. Since braking usually with decelerating and steering, so these two commands are strong correlation and have temporal characteristics. If attacker send fake steering command to make car turning inappropriately, the temporal characteristics of steering and braking sequence will be break and be detect out by our model.

Considering the real-time of vehicle detection, the LSTM model is designed as simple as possible to reduce the computational complexity. The input layer has one input neuron, the hidden layer has four neurons, and the output layer has one neuron. The cost function is mean square deviation used to minimize the mean square deviation between the predicted value and the marked value, and the smaller of the cost function, the better. Activation function and optimization function are set by experiments. The simulation tool is Pycharm using Keras and Tensorflow deep learning toolkit. We chose several common functions of the toolkit and check which activation function and optimization function will give the best result as shown in the Table 1. We can see the Nadam activation function and Elu optimization function give the lowest loss.

| Activation Function | Optimization Function | Loss    |
|---------------------|-----------------------|---------|
| Adam                | Selu                  | 0.1416  |
| Adam                | Softplus              | 0.1629  |
| Adam                | Softsign              | 0.1592  |
| Adam                | hard_sigmoid          | 0.1762  |
| Sgd                 | Elu                   | 0.2141  |
| RMSprop             | Elu                   | 0.1688  |
| Adagrad             | Elu                   | 0.2048  |
| Adadelta            | Elu                   | 0.2913  |
| Adamax              | Elu                   | 0.2474  |
| Nadam               | Elu                   | 0.1124  |

For the LSTM neural network, the most important issue is choosing an appropriate step which is the number of next commands is related to how many last comands. The Fig.6 shows the detection rate change with step from 1 to 12, and Fig.7 shows the Loss change with steps.
From above two figures, we can see that steps less than 6, the detection rates are almost 80%, and more than 6 the detection rate rise rapidly. But the detection rate did not increase significantly after 6 steps. The 7 steps even had lower detection rate than 6 steps. So, we chose 6 steps for our LSTM model and got 96.62% detection rate and 3.87% false positive rate. The detection target is basically achieved, and the detection time is in milliseconds.

6. Conclusion and Future Work
This paper focused on the knowledge representation and security of intelligent vehicles. A knowledge representation framework was proposed to offer uniform format for all kinds data of intelligent vehicle. With this framework the detection rules can be shared and inherited easily. And complex security experiences were stored in NK-DNA model. The LSTM neural network structure was used as an NK-DNA example to detect anomaly of CAN bus inside the vehicle. The simulation results shown our approach can detect malicious CAN command employing experiential knowledge.

7. Acknowledgments
This work was supported by the Scientific Research Foundation and Technology Support Plan Project of Sichuan Province as a part of the Project No.2016GZ0343, No.2019YFG0201, No.2019YFG0201, No. 2019YFG0398.

8. References
[1] K.Koscher, Czeskis A, Roesner F, et al. Experimental security analysis of a modern automobile [C]2010 IEEE Symposium on Security and Privacy. Piscataway, USA: IEEE Computer Society,2010:447-462
[2] Miller C, Valasek C. A survey of remote automotive attack surfaces [J]. BlackHat USA, 2014.
[3] Holle J, Platschek A, Sánchez J, et al. OVERSEE-A Secure and Open In-Vehicle ITS Station [C]. 19th ITS World Congress. 2012.
[4] E-safety vehicle intrusion protected applications (EVITA) Project [EB/OL]. http://www.evita-project.org.
[5] AUTOSAR documents [EB/OL] https://www.autosar.org/documents/.
[6] Groza B, Murvay S. Efficient protocols for secure broadcast in controller area networks [J]. Industrial Informatics, IEEE Transactions on, 2013, 9(4):2034-2042.
[7] Groza B, Murvay S, Van Herrewege A, et al.: Libra-can: a lightweight broadcast authentication protocol for controller area networks [J]. Cryptology and Network Security: Springer, 2012: 185-200.
[8] Woo S, Jo H J, Lee D H. A practical wireless attack on the connected car and security protocol for in-vehicle can [J]. Intelligent Transportation Systems, IEEE Transactions on, 2015, 16(2):993-1006.
[9] Chen Minxiou, Lin Bingyang. Design remote monitor system based on OSGi platform in vehicle gateway for vehicle network [C]. Proceeding 2nd International Conference on Next Generation Information Technology. USA: IEEE, 2011: 12-17.
[10] LIU Hui, LI Hui. A Scalable Anonymous Authentication Protocol for VANET [J]. Journal of Sichuan University. 2012, 44(3):131-136.
[11] Zhu Xiaoling, Hou Zhengfeng, Lu Yang. Secure decoding of vehicle black box based on secret sharing without trusted center [J]. Journal of electronic measurement and instrument. 2011, 25(3):279-284.
[12] ZHANG Wenbo, BAO Zhenshan, LI Jian. The Security Model for Vehicle Cloud Based on Trusted Computing [J]. Journal of Wuhan university. 2013, 59(5):438-442.
[13] WANG Jingxin, WANG Yue, GENG Junwei, et al. Secure and privacy-preserving scheme based on pseudonyms exchanges for VANET. Journal of Tsinghua University. 2012, 32(5):592-597.
[14] Haoxi Zhang#*, Fei Li, Juan Wang, Zuli Wang, Cesar Sanin, Edward Szczerbicki, Experience-Oriented Intelligence for Internet of Things, Cybernetics and Systems, 2017, 48(3):162-181
[15] Haoxi Zhang#*, Cesar Sanin, Edward Szczerbicki, and Ming Zhu, Towards neural knowledge DNA, Journal of Intelligent & Fuzzy Systems, 2017, vol. 32, no. 2, pp. 1575-1584
[16] Li Fei, Zhang H, Wang J, et al. Toward Intelligent Vehicle Intrusion Detection Using the Neural Knowledge DNA. Cybernetics & Systems, 2018:412-419
[17] Li Fei, Zhang H, Gao L, et al. A Collaborative Method for Multi-domain Information Security Mechanism of Intelligent Network Network Based on SOEKS. Cybernetics & Systems, 2019:1-8.
[18] Haoxi Zhang, Fei Li, Juan Wang, Zuli Wang, Lei Shi, Jun Zhao, Cesar Sanin & Edward Szczerbicki, Adding Intelligence to Cars Using the Neural Knowledge DNA, Cybernetics and Systems, 2017, 48:3, 267-273
[19] Greff K, Srivastava R K, Koutník J, et al. LSTM: A Search Space Odyssey [J]. IEEE Transactions on Neural Networks & Learning Systems, 2016, (99):1-11.
[20] Valasek, C., & Miller, C. (2013). Adventures in Automotive Networks and Control Units. Technical White Paper, 99. Retrieved from http://www.ioactive.com/pdfs/IOActive_Adventures_in_Automotive_Networks_and_Control_Units.pdf.
[21] Juan Wang, Haoxi Zhang, Fei Li, Zuli Wang, and Jun Zhao, Intelligent Vehicle Knowledge Representation and Anomaly Detection Using Neural Knowledge DNA, Frontiers in Cyber Security. FCS 2018. Communications in Computer and Information Science, vol 879. Springer, Singapore pp. 201–215.