The Monitoring System Based on a Multi-Agent Approach for Moving Objects Positioning in Wireless Networks

Oleksii Tohoiev[1][0000-0003-3465-7767], Ivan Burlachenko[1][0000-0001-5088-6709], Iryna Zhuravska[1][0000-0002-8102-9854], Volodymyr Savinov[1][0000-0002-0862-5879]

[1] Petro Mohyla Black Sea National University, 68 Desantnykiv str., 10, Mykolaiv, 54003, Ukraine
iryna.zhuravska@chmnu.edu.ua

Abstract. The problem of mobile device positioning system deployment is considered. Existing mobile positioning techniques that were intended for the moving object itself will not get on the Wi-Fi sniffer. A radio frequency propagation model method based on a multi-agent approach is proposed. Also, the router, host sniffer and phone can be in different network segments. This makes it possible to track any device that can access the Internet via Wi-Fi or Ethernet, like a mobile device in this approach, and to increase performance by 2% while reducing the number of calculations in the model. The sniffing software that implements the proposed methods is developed. It can be argued that the considered task is a supervised learning task with significantly accelerating the learning process.

Keywords: positioning for mobile devices, radiofrequency propagation, Wi-Fi N, hot spot mode, multi-agent analysis, sniffing software, accelerating the learning process

1 Introduction

The core concept of the Wi-Fi wireless network is the presence of the access point (AP), which connects to the terrestrial network infrastructure (Internet Service Provider) and transmits the radio signal. Usually, the AP consists of a receiver, a transmitter, a cable network interface, and firmware to quickly configure. A space area of 50–100 meters radius (called a hotspot or Wi-Fi zone) is formed around the AP, within which you can use a wireless network. The transmission distance depends on the transmitter power (which is programmatically configurated in some equipment models), the presence and characteristics of the interference, the type of antenna. Today, the 802.11n standard is widely used, which provides data rates of up to 320 Mbps [1].

Wi-Fi positioning techniques can be divided into two main groups. One is based on the Cartographic Catalog (CC) [2] of cartography and the other is based on modeling of radio wave propagation (RF model) [3]. The RF model determines the relationship between signal strength and distance. Determining the distance between known points and the mobile device allows using trilateration algorithms [4].

The purpose of the article is to study the operation of location algorithms' efficiency, and to develop a high-precision location algorithm of moving hosts in wireless networks.

2 Formal problem statement

Using RADAR technology, the mobile device uses the CC card of the required space [5]. The original CC map is formed from the coordinates, the CC measurements and the location of the mobile device [6]. The CC card can be based on calculations as well as physical measurements. The strength of the signal from each AP is compared with the corresponding indicators in the database, and the appropriate location is assumed. Transparent location fingerprinting uses a map of baselines. The reference point is a sequence of pairs (ss_j, c_j). Where ss_j is the set of signal strength measurements and c_j is the corresponding physical coordinates. The agreement of a new set of measurements of the signal strength *ss* is carried out by choosing *k* basis points closest to the obtained measurements, the average weight c_f which calculated using the formula (1):

$$c_f = \frac{\sum_{j=1}^{k} \frac{1}{d(ss_j, ss) + \varepsilon} c_j}{\sum_{j=1}^{k} \frac{1}{d(ss_j, ss) + \varepsilon}}, \tag{1}$$

where $d(ss_j. ss)$ is the Euclidean distance between two triples of APs and ε is a constant. The average error of this method is 1.78 m, but the maximum error can be up to 10 m.

We propose solving the problem of improving moving hosts' positioning in wireless network accuracy via the identification performed by a neuron network, depending on the types of tasks performed by devices.

3 Literature review

3.1 Technologies for positioning a mobile device based on data mining

The location of the object can be determined both in global and local coordinates. To date, there are widespread global positioning systems GPS, GLONASS, Galileo, BeiDou [7].

The main advantages of these technologies are its large coverage and high enough accuracy for determining outdoor locations. The main disadvantages of such systems include the dependence on weather conditions and the inability to use indoors with a lot of noise both from various equipment inside the building and from the building itself [8].

For positioning in local coordinates (to determine the movement of customers, intra-warehouse logistics, etc.), equipment with built-in Wi-Fi, RFID, Bluetooth modules, as well as accelerometers, compasses, smartphones antennas are used [9, 10]. However, data from these modules are hampered for both registration and analysis.

Probabilistic technologies use the probability distribution of signal strength for each reference point with indicators above the average. For example, the Ekahau Positioning Engine (EPE) uses two evaluation functions to coordinate measurements with a database using the formula (2).

$$C_{xy}(S_i) = \begin{cases} 0, \text{if } r(\theta) + re(\theta) \le d(S_i, P) \\ f(\theta)e^{-\lambda\alpha\beta}, \text{if } r(\theta) - re(\theta) < d(S_i, P) < r(\theta) + re(\theta), \text{and } \theta \in [0, 2\pi]. \\ f(\theta), \text{if } r(\theta) - re(\theta) \ge d(S_i, P), \text{and } \theta \in [0, 2\pi] \end{cases} \tag{2}$$

The first function is calculated by the kernel method, and the second function is calculated by the histogram method [11].

The HORUS system is based on Bayesian logic [12]. Its main feature is data grouping, which reduces the computational cost and time required to determine the location of a moving device connected to AP. This approach is also probabilistic. Each reference point contains a sample of 240 measurements. Samples are stored in the form of histograms, each of which combines all APs into a joint distribution.

It should be noted that the need for multiple measurements during continuous monitoring of users to determine their positions leads to high power consumption and, ultimately, to a significant reduction in the operating time of a mobile device without recharging [13, 14]. In this case, it is advisable to redistribute the tasks of continuous monitoring for execution on slower cores of a mobile device with low energy consumption [15].

3.2 Positioning based on modeling the propagation of radio waves

The purpose of this simulation is to express the mathematical relationship between the distance from the receiver to the transmitter and the signal strength. The mathematical expression is obtained from the polynomial regression of the third order. The main advantage of this technology is in positioning speed. In addition, an important point is that APs are stationary with known coordinates. An actual problem is the development of methods for determining the location of a user based on a combination of characteristics of signals from APs.

However, regression requires a large amount of accurate information about signal strength over a fairly long time. This technique provides positioning accuracy from 1 m to 3 m. An integral quadratic quality criterion is used to evaluate the effectiveness of positioning technologies [4].

Several dozen measurements are required to determine the relationship between distance and signal strength. It follows that this model is not fully dynamic.

Studies show that trilateration-based systems are characterized by lower location errors compared to the Bayesian approach [16]. However, it can be seen from the presented results that on average in 10% of cases a mobile object will be unprotected when a user works on a corporate network [17].

To monitor numerous objects authorized in a wireless corporate network, a multi-agent approach is often used [18, 19]. However, in this case, it is important to take into account the multipath nature of the radio channel in calculating the characteristics of the signal during direct communication of moving objects and communication through access points [20].

In the last decade, the localization of Wi-Fi-based objects has become one of the most popular solutions and is considered the most promising for the study of the raised issues in both scientific and industrial communities [21].

4 The modified method of a moving object positioning

4.1 Wi-Fi hot spot modes

The Wi-Fi net diagram contains at least one AP and can be easily scaled.

It is also possible to connect two clients in Ad-hoc mode when the AP is not in use and clients are connected to network adapters directly. The AP transmits its network identifier (SSID) via special signal packets at a rate of 0.1 Mbps every 100 ms. Therefore, 0.1 Mbps is the lowest data rate for Wi-Fi. Knowing the SSID of the network, the client can determine whether a connection to this AP is possible. When two hotspots with identical SSIDs are in range, the receiver can choose between them based on the signal level data. The Wi-Fi standard gives the customer full freedom in choosing the criteria for connection.

However, the standard does not describe all aspects of wireless LAN construction. Therefore, each manufacturer of the equipment solves this problem in its own way, applying the approaches that it considers to be the best from one point or another. Therefore, there is a need to classify ways to build wireless LANs.

By the pooling of APs into a single system, it's possible to distinguish:

- autonomous APs (standalone, decentralized, smart);
- manageable APs (controller-based, centralized);
- uncontrolled but not autonomous (cloud-based).

By the way of radio channels organization and management, wireless LANs can be distinguished:

- with static radio channel settings;
- with the dynamic (adaptive) tuning of radio channels;
- with the multi-layered structure of radio channels.

4.2 The method of multi-agent analysis

To view the specific traffic data that the agent $A^{s}_{m(i)}$ sends to the network, you will need more agents $A^{sn}_{m(i)}$ with a special sniffing role. Sometimes multi-agent monitoring software can be installed on the agent's host H_{RoA} (3):

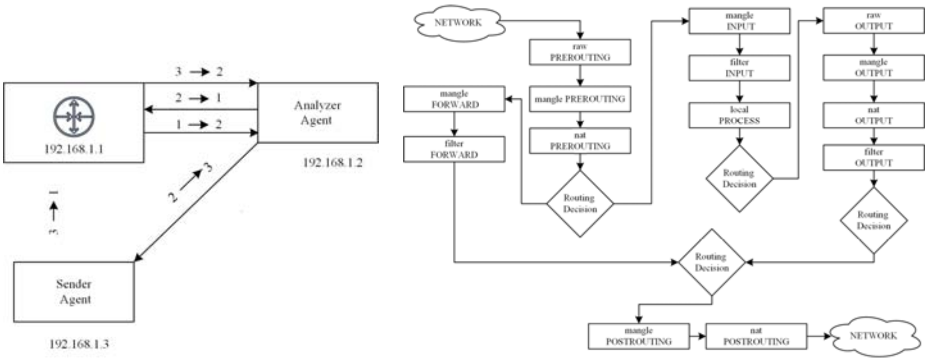
$$RoA = \left\{ j \in RoA \mid A^{j}_{m(i)} \right\}, \tag{3}$$

where *RoA* is a set of agents' behaviors. This is the most reliable way, but it is not always available. If the H_{RoA} has a wireless network connection, then you can use additional agents' devices $m(i)$. That means, connect it via Wi-Fi to a router or an agent's host with a sniffing role, or if there is no access to a router with Wi-Fi, connect it to an unprotected Wi-Fi network and analyze all traffic's data from another movable agents' hosts (4):

$$RoA_B = \{m(i-k_{lb}), \ldots m(i), \ldots m(i+k_{rb}) | (k_{lb}, k_{rb}) \in B\}, \tag{4}$$

Consider the most typical case a corporate network controlled by a regular Wi-Fi AP. Most corporate APs are running widespread OS, which can be accessed by agents of the multi-agent monitoring system (MAMS) without problems via Telnet protocol.

This experiment will investigate the process of the traffic's redirection from movable agents (or from any other host on the network) to an agent with a sniffing role to analyze the H_{RoA} under monitoring (Fig. 1, a).



a. b.

Fig. 1. Dataflow between movable agents at the subnet (a) and *iptables* routing diagram (b)

A network router as an agent was configured on the Linux base. The agent had a setup API available onboard and had network IP address 192.168.1.1. An Analyzer Agent includes the *iptables* and *iproute2* utilities and the *tcpdump* sniffer, with which you can solve almost all network tasks. An Analyzer Agent has an IP address: 192.168.1.2. The device H_{RoA} under investigation is a Sender Agent which connected to the network and has IP address 192.168.1.3.

The research plan is using *iproute2*, we create a separate routing table on the router and a rule that will use it for all traffic from the moving agents. We indicate in the table the default gateway – the host with the sniffing agent's role. Two rules with the same priority are not allowed. First, you need to check if there is another rule with the same priority: *ip rule list.*

It is advisable to choose a smaller number for priority – the lower the value, the higher the priority. The first four were busy. You can select any unused table *($TBL)*. Whether the table is used can be viewed using the *ip route list table $TBL* command. A host with a sniffing agent's role should redirect all the traffic under investigation to the Network. NAT can handle this easily. Further, this traffic will easily go through the same router and go to its destination – because now IP packets have a different sender address and it won't get into our routing table. That would seem to be all. You can run an Analyzer Agent on the H_{RoA} and filter traffic by IP address 192.168.1.3.

But this scheme has a significant drawback: the traffic that was intended for the router agent itself will not get on the sniffer. To solve this problem the Sniffing Agent role was used (Fig. 2). Judging by the scheme from the *iptables* by MAMS, so that as a result of routing this traffic is redirected (to the right branch), you need to change its destination address in *PREROTING*: *192.168.1.3.*

It should be noted that on the sniffer agent at the H_{RoA} host, traffic will fall under masquerading (*MASQUERADE*), that is, the source address will also change for packets. As a result, packets should move according to the following scheme in Fig. 2, b.

However, this option will not work if the H_{RoA} hosts are in the same segment, and the router acts as a network bridge. Then, according to the routing table on the Analyzer Agent's host, the reverse traffic will be sent directly to the recipient, and the recipient's OS should ignore it, because the sender's address will not be the one with which the connection is established.

The proposed way out was implemented in the RoA_B module of MAMS. According to k_{lb}, k_{rb} indices of range with agents' roles RoA_B redirects these packets to the router again, similar to how the redirection to the sniffer was done.

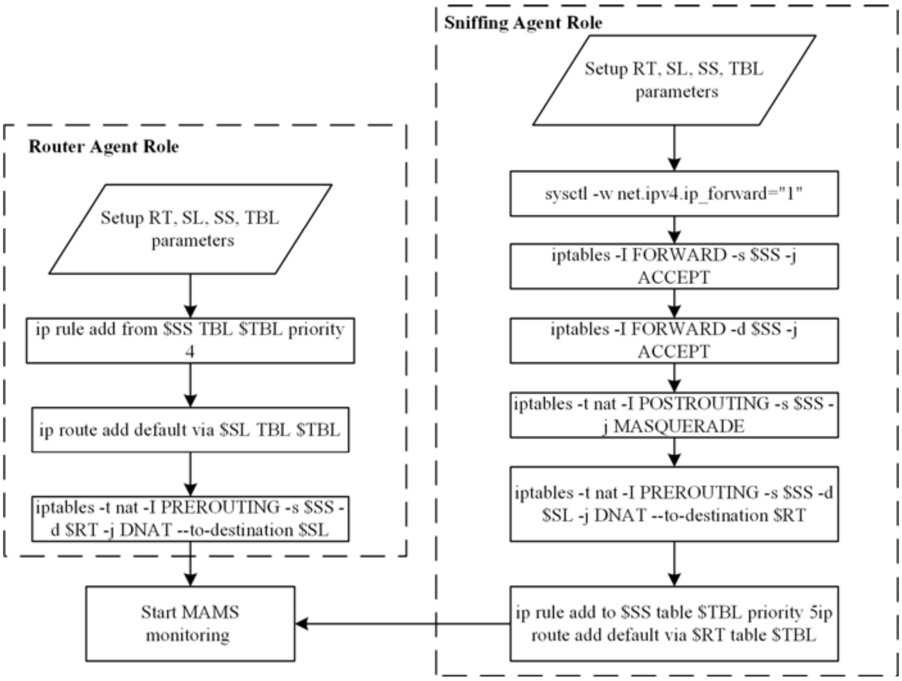
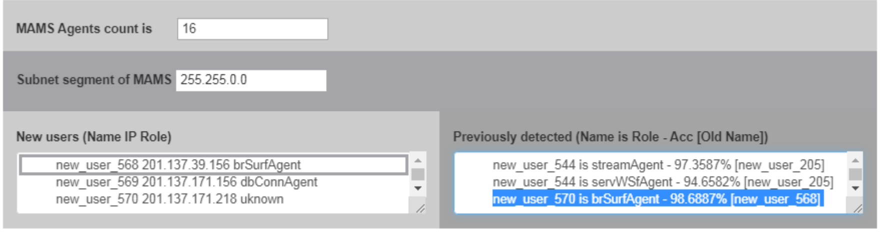

Sniffing Agent Role
Setup RT, SL, SS, TBL parameters
sysctl -w net.ipv4.ip_forward="1"
iptables -I FORWARD -s $SS -j ACCEPT
iptables -I FORWARD -d $SS -j ACCEPT
iptables -t nat -I POSTROUTING -s $SS -j MASQUERADE
iptables -t nat -I PREROUTING -s $SS -d $SL -j DNAT --to-destination $RT
ip rule add to $SS table $TBL priority 5ip route add default via $RT table $TBL
Router Agent Role
Setup RT, SL, SS, TBL parameters
ip rule add from $SS TBL $TBL priority 4
ip route add default via $SL TBL $TBL
iptables -t nat -I PREROUTING -s $SS -d $RT -j DNAT --to-destination $SL
Start MAMS monitoring

Fig. 2. Multi-agent Monitoring System (MAMS) for the network traffic analysis

When MAMS receiving this problematic traffic, the H_{RoA} with Router Agent Role, using the state-determining mechanism (*nf_conntrack* module), recognizes the connections changed with DNAT in it and replaces the packet receiver address with the original one. Next, the packets are sent to the recipient as if nothing had happened.

5 Experiments and results

The previous methodology that allows analyzing network properties of all networks' agents was described. So that gives the ability to investigate the problem of automatically classifying agents into many categories during establishing multiple connections in different networks' segments (Fig. 3).



MAMS Agents count is 16

Subnet segment of MAMS 255.255.0.0

New users (Name IP Role)

new_user_568 201.137.39.156 brSurfAgent
new_user_569 201.137.171.156 dbConnAgent
new_user_570 201.137.171.218 uknown

Previously detected (Name is Role - Acc [Old Name])

new_user_544 is streamAgent - 97.3587% [new_user_205]
new_user_544 is servWSfAgent - 94.6582% [new_user_205]
new_user_570 is brSurfAgent - 98.6887% [new_user_568]

Fig. 3. The user interface of the proposed MAMS during the identification new user's role and network's positioning

As the source data for machine learning were provided a database containing ~10 thousand network packages from MAMS, divided into 8 categories according to the network's activities of agents which were positioned up to 256 network segments. Payloads content of network packages is divided into categories, the names of which were hidden intentionally, because this data cannot be shown, rather unevenly.

Here are features which an agent gets from a payload of the H_{RoA} devices (5):

$$A_H^{PL} = \{A_{DA}, A_{DO}, A_{IDs}, A_{DAl}, A_{DSD}, A_{NC}, A_{CD}\}, \tag{5}$$

A_{DA} describes device attributes: operating system information, hardware and software versions, battery level, signal strength, free storage space, browser type, names and types of applications and files, and plugins. A_{DO} is device operation: information about the device's operation and behavior during use. A_{IDs} describes IDs: the individual IDs, device IDs, and other IDs of the games, apps, or accounts you use, and the IDs of the devices or other unique software IDs for a single device or account. A_{DAl} are device alerts: Bluetooth alerts, as well as information on close Wi-Fi hotspots, beacons, and mobile towers. The feature A_{DSD} is used for storing device settings data. The information you allow us to retrieve through your device settings, such as access to GPS data, cameras, or snapshots. A_{NC} is network and connectivity: information about your network operator or ISP, language, time zone, mobile phone number, IP address, connection speed, and, in some cases, information about other networks H_{RoA} nearby. A_{CD} is cookie data: the cookie data stored on H_{RoA}, including cookie IDs and cookie settings.

5.1 Classification of agent category according to subnet position using the LSTM model

After the supervised training for $A_{m(i)}^{sn}$ agents is defined, its solution can be share between $A_{m(i)}^{sn}$ on the different H_{RoA}. This task for other agents with similar roles is currently being solved by the following algorithm in Fig. 4.

The essence of the task of classifying an agent's category classification when agents appear in another network segment after its preliminary preparation and cleaning is to compile a dictionary of all the features available in A_H^{PL} payload texts, replace each category with a number – a unique word number in the dictionary, align the length of each category and network segments to the needed size (usually the number of elements in the longest feature's vector), and then any classification algorithm can be applied to the data presented in this form. In our case, the raw data was presented in the *MariaDB* database management system (DBMS). These data were successfully uploaded to the *Pandas DataFrame*.

At the preprocessing stage, the data have been analyzed: looked at data distribution, understand whether the data that we will analyze is missing information, in what form these data are presented. It is necessary to convert data into a single comparable form. In our case, a dataset of the subnet addresses for the distributed agents' applications $A^{s}_{m(i)}$ was formed and the data specific traffic by category was evaluated using *df.head()* function. Various special characters that needed to be removed from the *DataFrame* were normalized using *pymorphy2*.

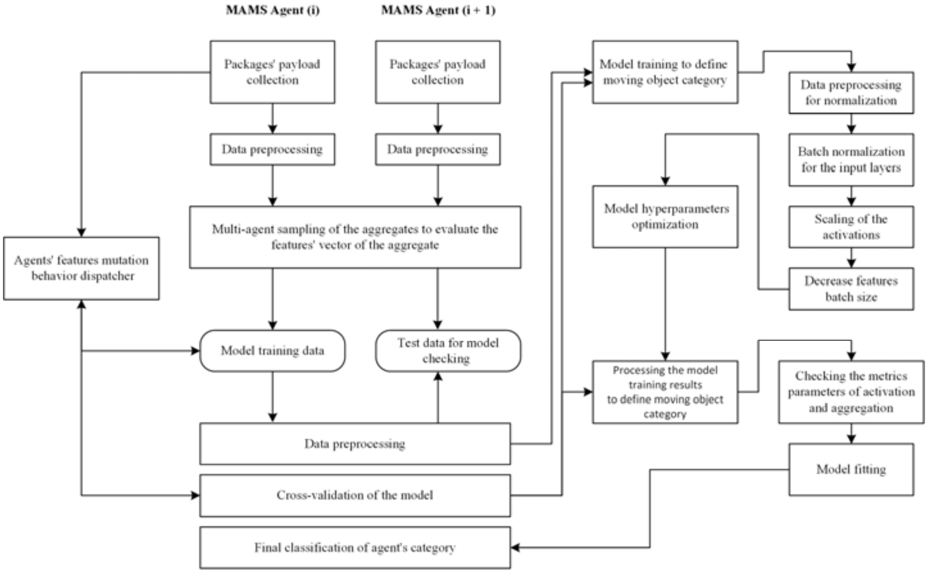

MAMS Agent (i)
MAMS Agent (i + 1)
Packages' payload collection
Packages' payload collection
Data preprocessing
Data preprocessing
Multi-agent sampling of the aggregates to evaluate the features' vector of the aggregate
Agents' features mutation behavior dispatcher
Model training data
Test data for model checking
Data preprocessing
Cross-validation of the model
Final classification of agent's category
Model training to define moving object category
Data preprocessing for normalization
Batch normalization for the input layers
Scaling of the activations
Decrease features batch size
Model hyperparameters optimization
Processing the model training results to define moving object category
Checking the metrics parameters of activation and aggregation
Model fitting

Fig. 4. Modified MAMS agents' algorithm for supervised learning to identify the category of moving objects

DataFrame's column contains text data with features of the H_{RoA}. They must be replaced with numerical data, matching the name of each category with its unique number. Because of a lot of records, and their processing takes considerable time, then to speed up work in the future, periodically save changes to your *DataFrame* is recommended, for example, in pickle format. Data in a *DataFrame* are read quickly and conveniently at any time. So, before starting the training, it remains for us to convert all the A_{H}^{PL} features descriptions of the agent into a comparable form and break the data. The mixing of the rows of the entire *DataFrame* so that all categories of agent's roles fall into the training and verification data sets is highly recommended.

The training and test data sets will contain an array of numbers indicating the A_{H}^{PL} features descriptions of the agent (*payP1*, *payP2*) and the $A_{NC(SNA)}$ subnet address to which the agent refers (*payP3*). Accordingly, the data for training will be recorded in the variables *payP1_train*, *payP2_train*, *payP3_train*, and the data for testing the training algorithm will be recorded in the variables *payP1_test*, *payP2_test*, *payP13_test*. This stage takes an input array with descriptions of agent’s features strings, an array with a numerical designation of the $A^{s}_{m(i)}$ agent’s category labels and the variable needed to vary the amount of data for the test and verification. In our experimental investigation, 90% of the data for training and 10% for checking the result were used. For varying portions, agents' features mutation behavior dispatcher was implemented. The number of the categories of the agent’s behavior in the dictionary we are dealing with is 8 and the number of subnet addresses is up to 256.

To solve the categorization problem the agent logic using Keras model (Fig. 5) was proposed. To define the position of moving agent could be used only 1700, which is approximately 17% of the entire dataset of 10 thousand frames with a payload.

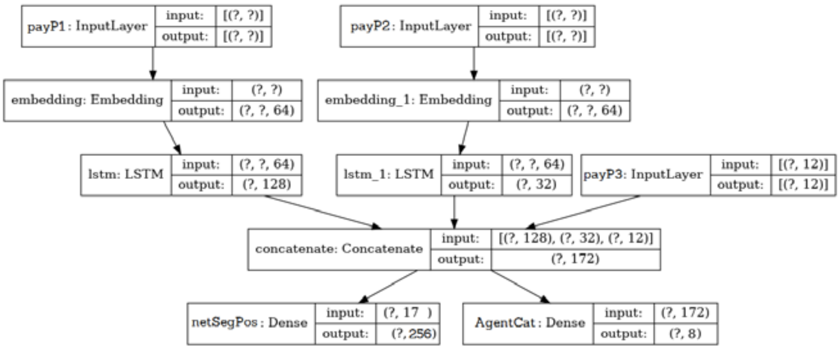


Fig. 5. Directed acyclic graph (DAG) of layers for the proposed agent’s behavior identification based on the subnet addresses *(netSegPos)* and category *(AgentCat)*

Usage of mutation behavior dispatcher as part of MAMS significantly reduces the number of calculations and significantly accelerates the learning process. Based on these considerations, training and test data were redefined. Cross-validation data is used to investigate whether your model over-fits the data.

To find the categorization problem solution, Long Short-Term Memory (LSTM) models were included as a layer of DAG. On our data, this model showed 98.7% accuracy in classification (Fig. 6).

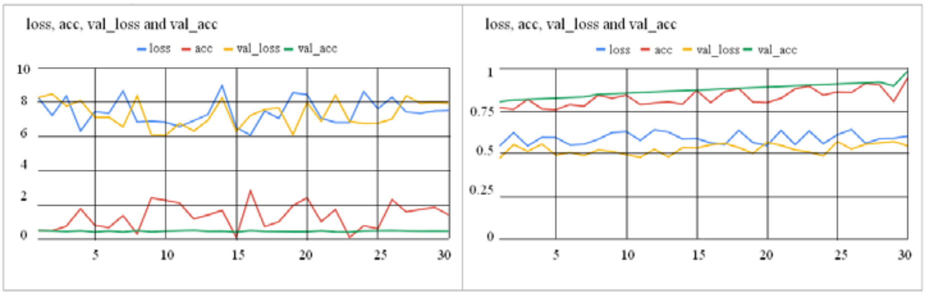


Fig. 6. The accuracy plot of the agent’s classification models

We trained the resulting model by writing the learning history into the history store with the predefined batch size that was 32 and 30 epochs. After the model was trained the proposed model accuracy has been validated.

6 Conclusion

Both *iptables, iproute2* agent behavior packages must be installed on both the router and agents' hosts, and agents with traffic filtering role and advanced routing support must be included in the kernel. Independent MAMS configuration of the router's kernel has decrease packets lost on 14% during the agent's data exchange and improve performance time by 70 ms faster.

Agents with router-, hostsniffer- and sender roles can be in different network segments. The main condition is that they are connected, and all traffic from the agents passed through the MAMS hosts.

Any other device that can access the Internet via Wi-Fi or Ethernet can act as an agent. Remember that even Wi-Fi protected networks can be unsafe if you are not sure that the AP is completely protected from unauthorized persons. If this is important, use data transfer protocols that support encryption: HTTPS, XMPP.

Of course, the application of this method to a computer with a full-fledged OS is easily monitored using *traceroute* (*tracert* on Windows), or simply because of a decrease in TTL packets, which is noticeable when the router pings.

The proposed MAMS provides the result of the correct definition of the category in 98.7% of cases using LSTM during of agent's behavior analysis.

References

1. Colbach, G.: The WiFi Networking Book: WLAN Standards: IEEE 802.11bgn, 802.11n, 802.11ac and 802.11ax. Independently Published (2019)
2. Longley, P.A., Goodchild, M.F., Maguire, D.J., Rhind D.W.: Geographic Information Science and Systems. John Wiley & Sons (2015)
3. Radio Propagation and Antennas: A non-mathematical treatment of radio and antennas. Cerwin S. AuthorHouse (2019)
4. Pantiukhin, A.R., Belyaev, A.S.: System of determination of objects location inside of premises. Int. Res. Jrnl. 10 (64): 81–84 (2017). doi: https://doi.org/10.23670/IRJ.2017.64.012
5. Bahl, V.: RADAR: An in-building RF-based user location and tracking system. In: Proc. of the 22nd Annual Int. Conf. on Mobile Computing and Networking (MobiCom 2016). https://www.youtube.com/watch?v=XI5-t5kKENk
6. Field-Map is a product of IFER – Monitoring and Mapping Solutions, Ltd. Praha, Czech Republic (2019). https://www.fieldmap.cz/download/FM_Catalogue_en.pdf
7. Santerre1, R., Pan, L., Cai, C., Zhu, J.: Single Point Positioning using GPS, GLONASS and BeiDou Satellites. Positioning. 5: 107–114 (2014). doi: 10.4236/pos.2014.54013
8. Rumiankov, D., Zhuravska, I., Solobuto, L., Musiyenko, M.: Reduction of noise similar to solar interference in computer networks based on Power Line Communication. In: Proc. of the IEEE 8th Int. Conf. on Intell. Data Acquisition and Adv. Computing Syst.: Technol.

and Applic. (IDAACS’2017), Bucharest, Romania, 21–23 Sept. 2017, vol. 1, pp. 215–221 (2017). doi: 10.1109/IDAACS.2017.8095079

9. Krainyk, Y., Davydenko, Y., Tomas, V.: Configurable Control Node for Wireless Sensor Network. In: Proc. of the IEEE 3rd Int. Conf. on Adv. Information and Communic. Technol. (AICT’2019), Lviv, Ukraine, 2–6 July 2019, pp. 258–262 (2019). doi: 10.1109/AIACT.2019.8847732
10. Dabove, P., Di Pietra, V., Lingua, A.M.: Positioning Techniques with Smartphone Technology: Performances and Methodologies in Outdoor and Indoor Scenarios. In: Smartphones from an Applied Research Perspective. Nawaz Mohamudally (2017). doi: 10.5772/intechopen.69679
11. Krzysztofik, W.J.: Radio Network Planning and Propagation Models for Urban and Indoor Wireless Communication Networks. In: Antennas and Wave Propagation (2018). doi: 10.5772/intechopen.75384
12. Youssef, M., Agrawala, A.: The Horus WLAN location determination system. In: Proc. of the 3rd Int. Conf. on Mobile Syst., Applic., and Services (MobiSys). Seattle, Washington, USA (2005). doi: 10.1145/1067170.1067193
13. Huber, D.: Background Positioning for Mobile Devices – Android vs. iPhone (2019), https://www.snet.tu-berlin.de/fileadmin/fg220/courses/WS1011/snet-project/background-positioning_huber.pdf
14. Obukhova, K., Zhuravska, I., Burenko, V.: Diagnostics of power consumption of a mobile device multi-core processor with detail of each core utilization. In: Proc. of the IEEE 15th Int. Conf. on Adv. Trends in Radioelectronics, Telecomm. and Computer Engin. (TCSET), Lviv, Ukraine, 25–29 Feb. 2020 (2020)
15. Zhuravska, I.M., Koretska, O.O., Musiyenko, M.P., Surtel, W., et al.: Self-powered information measuring wireless networks using the distribution of tasks within multicore processors. In: Proc. of SPIE: Photonics Applic. in Astr., Communic., Ind., and High Energy Physics Exper., Wilga, Poland, 28 May – 6 June 2017, vol. 10445, pp. 1–13 (2017). doi: 10.1117/12.2280965
16. Markin, D.O., Makeev, S.M.: Mobile device location system model based on Monte-Carlo method. Izvestiya TulGU. Technical science. 2: 150–165 (2016)
17. Burlachenko, I., Zhuravska, I., Davydenko, Ye., Savinov, V.: Vulnerabilities analysis and defense based on MAS method in fast dynamic wireless networks. In: Proc. of the Wireless Syst. within of the IEEE 4th Int. Conf. on Intell. Data Acquisition and Adv. Computing Syst. (IDAACS-SWS), Lviv, Ukraine, pp. 98–102 (2018). doi: 10.1109/IDAACS-SWS.2018.8525692
18. Burlachenko, I., Zhuravska, I., Musiyenko, M.: Devising a method for the active coordination of video cameras in optical navigation based on multi-agent approach. Eastern-European Jrnl. of Enterprise Technol. 1, 9 (85): 17–25 (2017). doi: 10.15587/1729-4061.2017.90863
19. Burlachenko, I.: Management of energy efficient distributed computer systems with self-contained remote modules using multi-agent system. In: Proc. of the IEEE 35th Int. Conf. on Electronics and Nanotechnology (ELNANO), Kyiv, Ukraine, 1: 512–514 (2015). doi: 10.1109/ELNANO.2015.7146940
20. Rida, J.F.A.: Improvement for performance radio frequency in wireless communication based on impulse signal. Indonesian Jrnl. of Electr. Engin. and Computer Sci. 18 (2): 903–916 (May 2020). doi: 10.11591/ijeecs.v18.i2.pp903-916
21. Wu, C., Yang, Z., Liu, Yu.: Wireless Indoor Localization: A Crowdsourcing Approach. Springer (2018)