Use of Adaptive Boosting Algorithm to Estimate User's Trust in the Utilization of Virtual Assistant Systems

[1]Akazue Maureen, [2]Onovughe Anthonia, [3]Edith Omede,
[1,2,3] Department of Computer Science, Faculty of science, Delta State University Abraka

[4]Hampo, John Paul A.C.
[4] Computer Science Department, Federal University of Technology, Owerri

Abstract:- User trust in technology is an essential factor for the usage of a system or machine. AI enabled technologies such as virtual digital assistants simplify a lot of process for humans starting from simple search to a more complex action like house automation and completion of some transitions notably Amazon's Alexa. Can human actually trust these AI enabled technologies? Hence, this research applied adaptive boosting ensemble learning approach to predict users trust in virtual assistants. A technology trust dataset was obtained from figshare.com and engineered before training the adaptive boosting (AdaBoost) algorithm to learn the trends and pattern. The result of the study showed that AdaBoost had an accuracy of 94.31% for the testing set.

***Keywords*:-** *Machine Learning, Ensemble Model, Predictive Model, Trust, Intelligent Virtual Assistants*

I. INTRODUCTION

Intelligent machines, and more broadly, intelligent systems, are becoming more prevalent in people's daily lives. Despite major advances in automation, human supervision and intervention are still required in almost every industry, from manufacturing to transportation to crisis management and healthcare [1]. AI is becoming more prevalent in various fields in the world notably in the medical industry. However, there are certain worries about A.I.'s ability to make critical decisions in a way that humans would consider fair, to be aware of and aligned with human values that are relevant to the problems being addressed, and to explain its thinking and decision-making. At its core, AI produces automated decisions, which can lead to a decision-making bias because physicians are more likely to trust diagnostic test results generated by AI-driven machines without putting them through rigorous scrutiny [2].

Machine learning (ML) is a subset of data mining (DM). Data mining is the analysis of large data to discover hidden trends, patterns, and relationships, thereby developing a computer model to assist in decision-making. Data mining is also termed knowledge discovery in databases and datasets [3]; [4], and it constitutes a stage in Knowledge Discovery Database (KBB) process with other stages being data selection, data cleaning, and evaluation. Machine learning is the ability of a computer or a program to learn from experience concerning some tasks by the increase in the computer or program's performance.

Machine learning can be supervised, unsupervised or reinforcement learning. In supervised machine learning, models can be built that learn from the data, thereby the model having the ability to guesstimate the pattern and relation for the outputs.

➢ *Many algorithms can be used during the development of a model in supervised learning, which is developed in different mathematical backgrounds. They are:*

- *Linear and polynomial regression,*
- *Basis function construction using adaptive modeling,*
- *Relevance vector machines using the Bayesian framework,*
- *Support vector machines,*
- *Regression trees using recursive partitioning or continuous class learning,*
- *Multilayer perceptrons using neural networks.*
- *Development of models using supervised learning is done following the procedure below:*

➢ *After Deciding on the Input Parameters, Gather a Decent Amount of Training and Test Examples;*

- *Analyze the correlation between the input parameters. Less correlated inputs will give more reliable results;*
- *Choose one of the algorithms from the above list and decide upon the other parameters required for that algorithm;*
- *Train the model using the training examples and the chosen algorithm;*
- *Lastly, check the reliability and accuracy by predicting the target values for the test dataset using the developed model and comparing them with the actual values.*

Ensemble techniques of machine learning will be applied in the development of the predictive model to estimate user trust. Ensemble machine learning as a machine learning technique employs the development of various models and strategically integrating the various model to solve a problem which most times are computational intelligence problems [5]. Multiple learning algorithms are combined to solve a single problem, hence having better prediction and classification performance, than the performance from a single algorithm. Ensemble learning has been applied to an eclectic array of topics in regression, classification, feature selection, and abnormal point reduction [5].

Trust plays an important role in the usage of technological devices, the growth, and advancement of a system [6]. Trust is a term coined in most literature of social sciences, however, it has been adopted by the sciences, especially in computing. Trust is a measure of reliability, utility, and availability; that improves the overall functionalities of technological systems like the quality of services, reputation, availability, risk, and confidence.

Human trust is essential for successful human-machine interactions and human trust can be divided into three kinds in the context of autonomous systems: dispositional, situational, and learned [7]. Situational and learned trust is based on a given situation (e.g., task difficulty) and experience (e.g., machine reliability), respectively. Dispositional trust refers to the component of trust that is dependent on demographics such as gender and culture, whereas dispositional and learned trust is based on a given situation (e.g., task difficulty) and experience (e.g., machine reliability). Situational and acquired trust factors "may alter throughout a single conversation" [7], whereas all of these trust factors influence how humans make judgments when engaging with intelligent computers.

Human trust has been attempted to be predicted using dynamic models based on human experience and/or self-reported behavior [8]. However, retrieving human self-reported behavior continually for use in a feedback control system is impractical. Although these measurements have been linked to human trust levels [9], they have not been investigated in the context of real-time trust sensing. This research work offers a model to predict user trust based on collected datasets on users' interaction with artificial intelligence (AI) enabled machines and devices.

Human-computer interaction is the interaction that exists between humans and computers. The human is referred to as the user while the computer can be referred to as a machine. [10] defined human-computer interaction (HCI) as a discipline concerned with the design, evaluation, and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them.

Human-computer interaction has a basic goal and a long-term goal. The basic goal is to improve the interactions between users and computers by making computers more usable and receptive to the user's needs. In the long run, HCI has a goal of designing systems that minimize the barrier between the human's cognitive model of what they want to accomplish and the computer's understanding of the user's task.

Personal assistants enabled with AI technologies are usually referred to as Intelligent Virtual Assistants (IVA) or as Virtual Personal Assistants (VPN) [11]. IVA is a software that uses human voice and rational data to give help by noticing inquiries in the usual dialect, making proposals, and accomplishing activities.

Virtual assistants whether software or hardware (as in the case of Personal Digital Assistant) is an agent with the sole goal of assisting the user [12]; this is similar to the case of assistants in real-life settings. Intelligent virtual assistants are sometimes capable of operating in pre-recorded voices. Some of the tasks an intelligent virtual assistant can do are: questioning (online search), controlling home automation devices (switching on the air conditioner remotely), media playback, emails, to-do lists, and appointments. Some other activities IVAs can do are making informed suggestions and cracking funny one-liner jokes.

Table 1. Some IVAS and their Developer

S/N	INTELLIGENT VIRTUAL ASSISTANT	DEVELOPER
	Bixby	Samsung
	Alexa	Amazon
	Siri	Apple
	Google Assistant	Google
	Cortana	Microsoft

➤ *Summarily, Intelligent Virtual Assistants Though Like Software Chatbots, are More than Chatbots in their Range of Operation and Service Coverage [13]. Hence, they:*

- *Do things for users by focusing on task completion*
- *Get what the user says thereby having intent understanding through the conversation between the user and the iva*
- *Get to know the user thereby learning the user's personal information and applying them.*

II. MATERIALS AND METHODS

Jupyter notebook and Python programming language was used to develop this model in association with machine learning and data science frameworks such as Scikit learn, Pandas, NumPy, Matplotlib, and Seaborn. Scikit learn was used for building the model; it has all the algorithms and techniques used in the model-building process. NumPy and Pandas were used for linear Algeria and Scientific computing of this research. Pandas were also used in the cleaning and engineering of the data. Seaborn and Matplotlib are also data science packages that were used for the visualization of the patterns in the dataset. The dataset used in this research was downloaded from figsshare.com.

A. Methodology

The existing system is a single model adopted by [14]. They implemented Decision Tree (DT) algorithm in building their model which gave them a precision of 92%.

The model presented in this study is an enhanced machine learning model for predicting users' trust in technology using a trust dataset from figshare.com. The architecture of the model is shown in **Figure 1**. The development process starts by pulling data from the storage repository from figshare.com, having a .csv format; which is an easy file format for python to import using the pandas' library. The data in the CSV file is then converted to vectors which is the format the model can be trained on, the hyperparameters are tuned till the model is ready for use, once the performance was sufficient the model was then saved to disk for prediction.

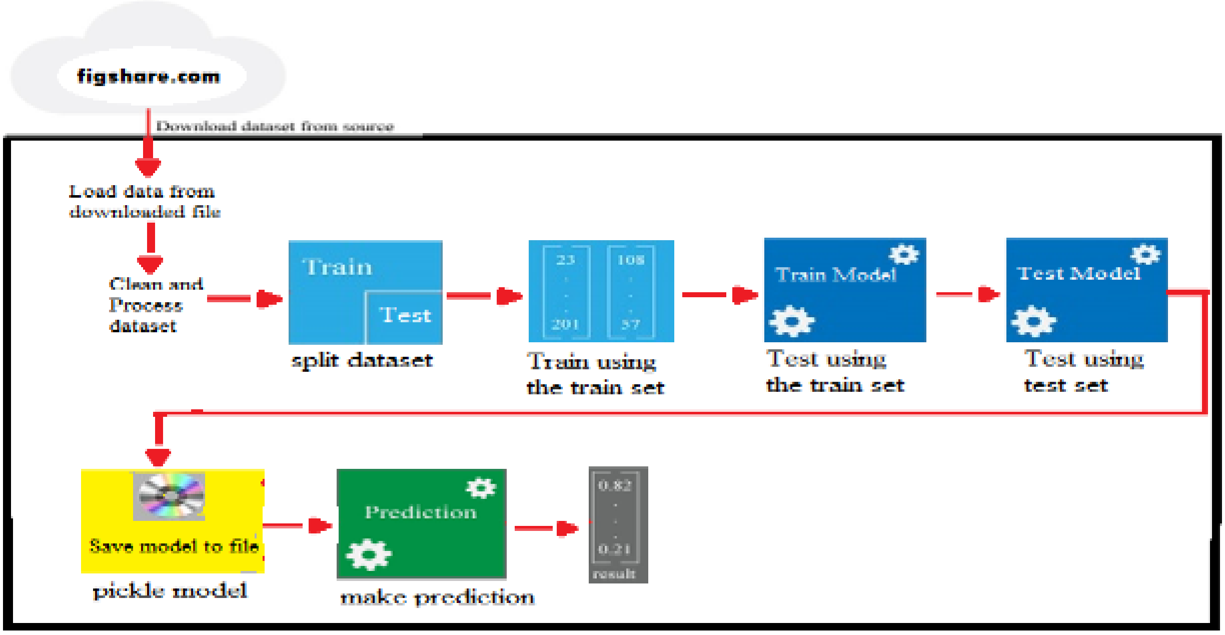


Fig 1 High Level Architecture of the Proposed Model

During the development of the model, the trust data was obtained from figshare.com and the fields were engineered to fit the proper predictive model of this research. Using the train-test-split algorithm of scikit-learn python framework, the dataset was split into 75% for training and 25% for testing, the dataset was then converted to vectors and trained using logistic regression from the sci-kit-learn library which was then exported for deployment and use outside the development environment, the model's hyperparameters were tuned for the model to achieve the best accuracy and this continues iteratively till the training is completed and the model is saved to storage which can be deployed on several platforms for predicting users trust in technology. The process of training the model is shown in **figure 2**.

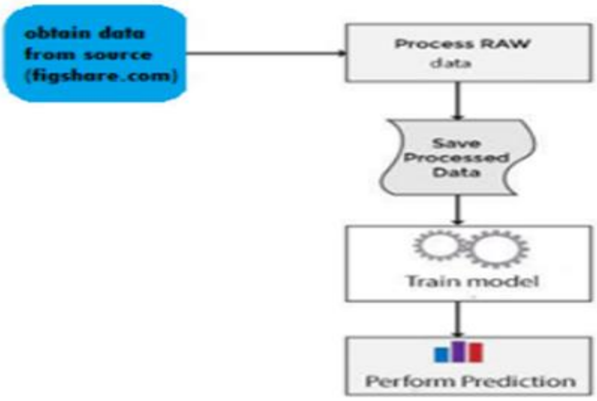


Fig 2 Training the prediction model to estimate user’s trust in technology

Different algorithms come into play to develop this trust predictive system. These algorithms are dataset cleaning, dataset engineering, model splitting, model building, and metric and evaluation algorithms.

B. Pseudocode For Data Cleansing

```
def dropna():
data = {"group": ["g1", np.nan, "g1", "g2", np.nan], "B": [0, 1, 2, 3, 4]}
    df = pd.DataFrame(data)
    grouped = df.groupby("group", dropna=False)
    result = grouped.indices
    dtype = np.intp
    expected = {
"g1": np.array([0, 2], dtype=dtype),
"g2": np.array([3], dtype=dtype),
        np.nan: np.array([1, 4], dtype=dtype),
    }
    for result_values, expected_values in zip(result.values(), expected.values()):
        tm.assert_numpy_array_equal(result_values, expected_values)
    assert np.isnan(list(result.keys())[2])
    assert list(result.keys())[0:2] == ["g1", "g2"]
```

C. Pseudocode For Data Splitting

```
    train_test_split(*arrays, test_size=None, train_size=None, random_state=None, shuffle=True, stratify=None)
```

➢ *Parameters*

- *Arrays: inputs such as lists, arrays, data frames, or matrices
- Test_size: this is a float value whose value ranges between 0.0 and 1.0. It represents the proportion of our test size. Its default value is none.
- Train_size: this is a float value whose value ranges between 0.0 and 1.0. It represents the proportion of our train size. Its default value is none.
- Random_state: this parameter is used to control the shuffling applied to the data before applying the split. It acts as a seed.
- Shuffle: this parameter is used to shuffle the data before splitting. Its default value is true.
- Stratify: this parameter is used to split the data in a stratified fashion.

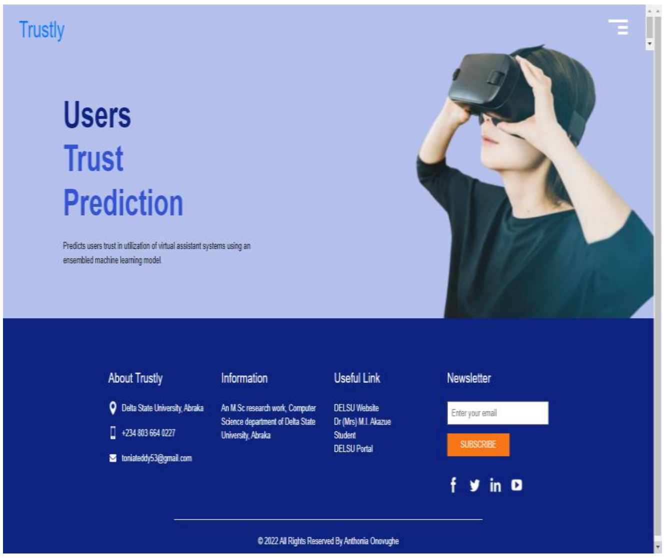
➢ *Algorithm*

- AdaBoost which is an acronym for Adaptive Boosting builds a strong classifier from multiple weak classifiers. The algorithm is simply below:

```
✓ Initialize the dataset and assign equal weight to each of the data points.
✓ Provide this as input to the model and identify the wrongly classified data points.
✓ Increase the weight of the wrongly classified data points.
✓ if (got required results)
✓ Goto step 5
✓ else
✓ Goto step 2
✓ end if
✓ End
```

III. IMPLEMENTATION

The pickled model was made available on visual studio code. It was de-pickled (loaded) using the pickle module in python programming language and deployed using flask framework. It consists of HTML, CSS, JavaScript and python programming language.



Users Trust Prediction

Predicts users trust in utilization of virtual assistant systems using an ensembled machine learning model.

Delta State University, Abraka
+234 803 664 0227
toniateddy53@gmail.com

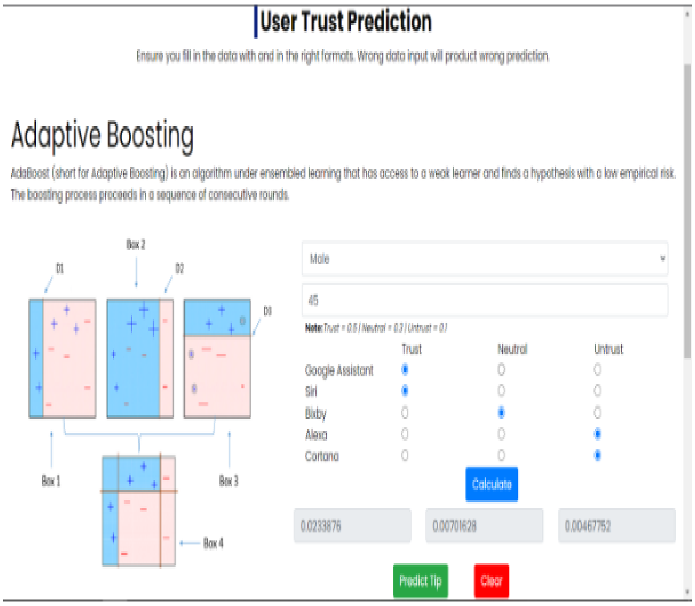
An M.Sc research work, Computer Science department of Delta State University, Abraka

DELSU Website
Dr (Mrs) M.I. Akazue
Student
DELSU Portal

© 2022 All Rights Reserved By Anthonia Onovughe

Fig 3 Trustly Home Page

Once the python web app is triggered using the command, *python app.py* on command prompt. The app runs on the localhost with address 127.0.0.1 and port 5000. When the localhost address is typed on the browser, the web app is made available as shown in figure 3. Figure 4 and figure 5 shows the input interface and the output interface respectively for prediction of users trust using the deployed web application.



User Trust Prediction

Ensure you fill in the data with and in the right formats. Wrong data input will product wrong prediction.

Adaptive Boosting

AdaBoost (short for Adaptive Boosting) is an algorithm under ensembled learning that has access to a weak learner and finds a hypothesis with a low empirical risk. The boosting process proceeds in a sequence of consecutive rounds.

Male

45

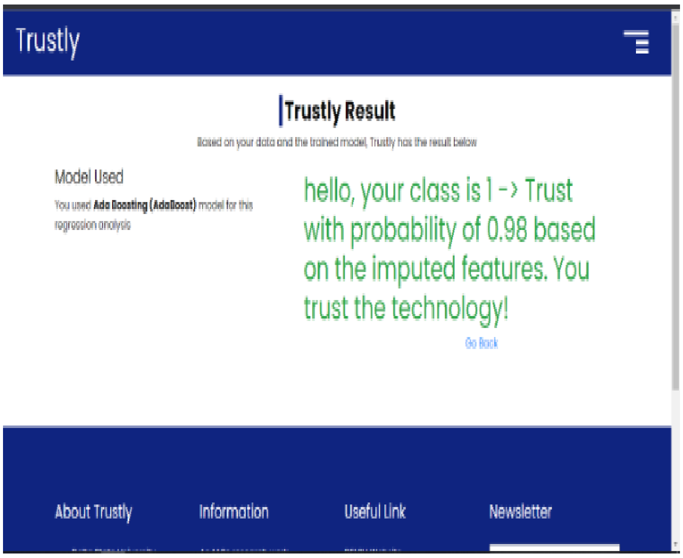
Note: *Trust = 0.5 / Neutral = 0.2 / Untrust = 0.1*

	Trust	Neutral	Untrust
Google Assistant	●	○	○
Siri	●	○	○
Bixby	○	●	○
Alexa	○	○	●
Cortana	○	○	●

0.0233876 0.00701628 0.00467752

Fig 4 Input Interface of the Deployed Model

Prediction is based on the values as computed from the selections made by the user, with respect to the listed intelligent virtual assistant. The final values for Trust, Neutral and Untrust is fed into the deployed machine learning model for the prediction of the user's trust. If the class after prediction is 1, then the user trust technology and the probability of trust is given. Also, if the outputted class after prediction is 0, then the user does not trust technology. The probability of trust which is close to 0 and less than 0.5 is also given.



Trustly Result

Based on your data and the trained model, Trustly has the result below

Model Used

You used **Ada Boosting (AdaBoost)** model for this regression analysis

hello, your class is 1 -> Trust with probability of 0.98 based on the imputed features. You trust the technology!

Fig 5 Output Interface of the Deployed Model

IV. PERFORMANCE EVALUATION

A. *Accuracy was Used to Evaluate the Performance of the Model.*

➢ *Below are vital performance metrics*:

- *True positive – this is the total number of correctly classified attacks. It is denoted as TP*
- *True negative – this is the total number of correctly classified non-attack. It is denoted as TN*
- *False positive – this is the total number of wrongly classified attack and it is denoted as FP*
- *False negative – this is the total number of wrongly classified non-attack and it is denoted as FN*

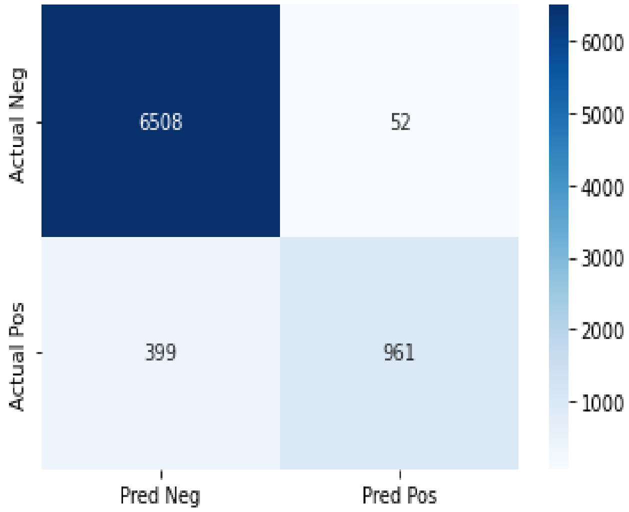


Fig 6 Model's Confusion Matrix

Accuracy – the accuracy of the classification is given as:

$$CA = \frac{TP + TN}{TP + TN + FP + FN} * 100$$

Accuracy gave 94.31%

Precision rate – percentage of all correctly classified of all attack packets, given as:

$$PR = \frac{TP}{TP + FP} * 100$$

Precision rate is 94.87%

Recall – percentage of all rightly classified attack in the dataset. This is given as:

$$RC = \frac{TP}{TP + FN} * 100$$

Recall is 70.66%.

V. CONCLUSION

This study proposes a predictive model to estimate users' trust in technology. The model is an enhanced machine learning statistical model, trained on the data from figshare.com where the model learns predictive features related to users' trust in technology.

The model was built using python and a couple of machine learning libraries, sci-kit-learn (aka sklearn) was used to build the machine learning model, NumPy for vector processing, and Pandas to handle CSV files where the training features are stored. Matplotlib and Seaborn were used for visualization. Pandas visualization module was also employed.

This study has shown that users' trust prediction concerning technology is better with an ensemble model using adaptive boosting algorithm which gave an accuracy of 94.31%.

This work can be applied to estimate user's trust in virtual intelligent assistants which are AI enabled technologies. The virtual intelligent assistants that were tested are Google Assistant, Siri, Cortana, Bixby and Alexa.

➢ *The Suggestions for Further Studies are as Follows:*

- *Further research can be carried out using the hybridization of ensemble learning and deep learning.*
- *Further research could seek to deploy the model into a mobile system.*

REFERENCES

[1]. Yue, W. and Fumin, Z. (2017). Trends in Control and Decision-Making for Human–Robot Collaboration Systems.

[2]. Dorado J., del Toro X., Santofimia M.J., Parren˜o, A., Cantarero, R., Rubio, A. and Lopez, J.C. (2019). A computer-vision-based system for at-home rheumatoid arthritis rehabilitation. International Journal of Distributed Sensor Networks.15(9). doi:10.1177/1550147719875649

[3]. Divya, K. and Srinivasan B. (2021). A TRUST-BASED PREDICTIVE MODEL FOR MOBILE AD HOC NETWORKS. International Journal on AdHoc Networking Systems (IJANS) Vol. 11, No. 3,

[4]. Sefer, K. and Yaseen, A. (2019). *Automatic Malware Detection using Data Mining Techniques Based on Power Spectral Density (PSD)*, International Journal of Computer Science and Mobile Computing **(IJCSMC)**, 8(3), PP27-30

[5]. Akazue, M. and Ojeme, B. (2014) Building Data Mining for Phone Business. Orient.J. Comp. Sci. and Technol;7(3)

[6]. Yan, L., and Liu, Y. (2020). An Ensemble Prediction Model for Potential Student Recommendation Using Machine Learning. *Symmetry*, *12*(5), 728. MDPI AG. Retrieved from http://dx.doi.org/10.3390/sym12050728

[7]. Kelvin, H. A. & Masooda, B. (2014). *Trust in Automation: Integrating Empirical Evidence on Factors That Influence Trust. Human Factors: The Journal of the Human Factors and Ergonomics Society, (), 0018720814547570–*. doi:10.1177/0018720814547570

[8]. Hussein, A., Elsawah, S. and Abbass, H. (2019). A System Dynamics Model for Human Trust in Automation under Speed and Accuracy Requirements. Proceedings of the Human Factors and Ergonomics Society 2019 Annual Meeting

[9]. Akash, K., Hu, W., Jain, N. & Reid, T. (2018). A Classification Model for Sensing Human Trust in Machines Using EEG and GSR. ACM Trans. Interact. Intell. Syst. 8, 4, 20 pages. https://doi.org/10.1145/3132743

[10]. Pravallika, B. (2018). LECTURE NOTES ON HUMAN COMPUTER INTERACTION. Information Technology, Institute of Aeronautical Engineering

[11]. Bhosale, V., Raverkar, D. and Kadam, K, (2021). GOOGLE ASSISTANT: NEED OF THE HOUR. Contemporary Research in India

[12]. Batura, A. (2019). INTEGRATING VIRTUAL ASSISTANT TECHNOLOGY INTO OMNI-CHANNEL PROCESSES. Bachelor's thesis Degree programme in business logistics, South Eastern University of Applied Science

[13]. Natale, S. (2020). To believe in Siri: A critical analysis of AI voice assistants. Loughborough University, UK.

[14]. El-Sayed, H., Ignatious, A.H., Kulkarni, P. and Bouktif, S. (2020). Machine learning based trust management framework for vehicular networks. Vehicular Communications 25, 100256. DOI: https://doi.org/10.1016/j.vehcom.2020. 100256