IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 12, No. 2, June 2023, pp. 943∼955
ISSN: 2252-8938, DOI: 10.11591/ijai.v12.i2.pp943-955

Modeling of an artificial intelligence based enterprise callbot with natural language processing and machine learning algorithms

Imad Aattouri[1], Hicham Mouncif[2], Mohamed Rida[1]
[1]Department of Math and Computer, Faculty of Sciences Ain Chock, University Hassan II, Casablanca, Morocco
[2]Multidisciplinary Faculty of Beni Mellal, University Sultan My Slimane, Beni Mellal, Morocco

Article Info

Article history:

Received Jan 28, 2022
Revised Sep 21, 2022
Accepted Oct 21, 2022

Keywords:

Callbots
Logistic regression
Machine learning
Natural language processing
Naïve Bayes
Random forest
Support vector machine

ABSTRACT

The management of customer services by telephone encounters several problems: an uncontrollable flow of calls, complicated resource management, a very high cost of service, and more. Opportunities to improve the quality of service, save time and money triggered the widespread implementation of artificial intelligence (AI) based callbot. This article outlines the straightforward workflow developed to model the architecture of the callbot. Therefore, several algorithms were evaluated and compared based on real knowledge of a call center of an insurance society. The algorithms considered are: k-nearest neighbours (KNN), support vector machine (SVM), random forests (RF), logistic regression (LR), and Naïve Bayes (NB). The comparison criteria are: correct responses, response time, accuracy, Cohen's kappa and F1 score using n-gram (1.1) and (2.2). The results obtained show that the SVM (accuracy=70.29%) presents the best results on all the comparison criteria. The comparison between the results of the human agents and the callbot shows an improvement in several levels: the cost savings are greater than 80% on all the tests carried out, the holding time decrease to 0 seconds, and the processing time (almost a third or more). The results obtained sufficiently meet the objectives of this project.

Corresponding Author:

Imad Aattouri
Department of Math and computer, Faculty of Sciences Ain Chock, Hassan II Univeristy
Casablanca, Morocco
Email: imad.aatt@gmail.com

1. INTRODUCTION

Many industrial and service companies nowadays offer their customers a remote service by telephone. Thus, the customer service companies are more and more present to meet the needs of the customers and with it was born a new concept of call center agent: the callbot. This concept is part of the technology family used to streamline communication with customers, such as voice agents, phonebots, conversational agents, and more. On other hand, several difficulties are identified in the management of customer services, among which we find: a large number of customer calls which leads to an enormous cost for the treatment, and a big difficulty to provide a 24/7 service with good quality. The callbot is a less well-known solution compared to another variant of the automatic system (chatbots [1]-[10]). This technology is an artificial intelligence (AI) that can manage a dialogue with a customer during a telephone call, to meet his need, and to solve it autonomously, 24 hours a day, without waiting time. This article outlines the straightforward workflow developed to model the callbot architecture. In this context, the machine has the role of understanding the dialogue and interacting with the client. This approach requires several designs and implementation axes, namely understanding human language and responding in the same way. The rest of this article is organized in the following order: section 2 presents reviews of related works. Section 3 contains the research methodology and details the proposed architecture. The result and discussion are presented in section 4. Section 5 contains a conclusion.

2. RELATED WORKS

This paper is a continuation of our previous works [11] and [12] concerning the realization of our AI-based callbot project. In the literature, most of the research papers deal with the chatbot problem. The researchers are more and more interested in callbots in previous years. The work [1] listed 10 machine learning algorithms (MLA) for training chat-bots, disclosed the information technology (IT) architecture of a chatbot platform using the natural language understanding (NLU) engine learning Bayesian method, and the results of testing this chatbot by users show a good satisfaction (75% of users are convenient with chatbot conversation). The article [2] gives a formal process description for realizing a chatbot, and realized a prototype tool to transform a process model in business process model and notation (BPMN) into a chatbot, defined in artificial intelligence marking language (AIML). The study conducted in the paper [3] made a comparison between two models based on sequence-to-sequence and AIML for building chatbots. The result obtained concludes that the sequence-to-sequence model had a better information retrieval rate while the AIML chatbot ensured better task completion rate and user satisfaction. Augello *et al*. [4], created a social chat bot model based on the social-AIML (SAIML), the proposed architecture allowed a more exact interpretation of user sentences thanks to the highlighting of social practice in the deliberative process of an agent. In order to improve the overall recommendation process, the authors of the article [5] have simplified the human-chatbot conversation with the conversational parameters supported by default. The proposed approach aims to avoid inconsistencies during the interactions with the chatbot. The project presented in the work [6] deals with an investigation about the use of chatbots to provide negotiation facilities and the incorporation of chatbots into an open learner modeling environment. Shalaby *et al.* [7], describes the main steps for developing a conversational virtual agent to understand and respond to complaints related to vehicle equipment. The results obtained show a precision that adapts better with a large volume of features up to 30% more accurate and is better at understanding user utterances with domain-specific entities. Cerezo *et al.* [8] presents the implementation of a chatbot developed for the Pharo software ecosystem. The chatbot includes several components: the discord application programming interface (API), term frequency (TF), and inverse document frequency (IDF) algorithms to perform sentence classification and key-concept collection respectively, and an expert recommendation system. The system was tested by the Pharo Community but the conversational behavior of the chatbot was not able to follow users' expectations. The work [9] develops an architecture for the Messenger chatbot using Amazon Web Services, the architecture has proven to be extensible and scalable.

One of the important phases in developing the architecture of our callbot is to define the decision algorithm module. In the following, we presented a comparative study of MLA. The MLA find their applications in several areas, namely: text classification [13]-[17], medical diagnosis [18], pollution prediction [19], spam email detection [20], plant disease identification [21], and stock daily trading [22]. For example, The paper [13] describes the use of the KNN algorithm with the TF-IDF method for text classification. The results obtained show that this combination proved to be a good choice with changes in their implementation. The work [14], presents a study of a BBC news text classification system. The algorithms covered are k-nearest neighbor (KNN) , random forest (RF), and logistic regression (LR). In this experimental the TF-IDF vectorizer feature and LR classifier attains the highest accuracy of 97% for the data set. The RF classifier gave an accuracy of 93%. With an overall accuracy of 92% The KNN was the algorithm with the least accuracy. In terms of all parameters The LR classifier gave a performance as expected. The study [15] shows that the effectiveness of the classifiers based on different training text corpuses is distinct and deduce that classifier performance is relevant to its training corpus in some degree, and good or high-quality training corpuses may derive classifiers of good performance. The authors of the paper [18] compare seven ML algorithms: LR, KNN, support vector machine (SVM), Naïve Bayes (NB), decision tree (DT), RF, AdaBoost (AB) on the Pima Indian Diabetes (PID) dataset to predict diabetes. They found that the model with LR and SVM works well on diabetes prediction. The authors of the study [19] tested 12 MLA to predict costs and carbon dioxide emission in an integrated energy-water optimization model and considered four indices to examine the prediction accuracy of the algorithms. Meanwhile, the light gradient boosting machine and extra tree algorithms enjoyed higher prediction accuracy in this research than other algorithms. To detect spam emails, the work [20] proposes a hybrid bagging approach that implements NB and J48 DT MLA. The proposed approach achieved 88.12% of overall accuracy with the hybrid bagged approach implementation. The contribution of the paper [22] is to make a comparison between the trading performance of the deep neural network (DNN) algorithms and traditional MLA in the Chinese stock market and the US stock market. The experimental results in S&P 500 index component stocks (SPICS) and 185 CSI 300 index component stocks (CSICS) show that some traditional MLA have better performance than DNN algorithms in most directional evaluation indicators. The paper [23] gives a systematic review of MLA in recommender systems which can help application developers to deal with the algorithms, their types, and trends in the use of specific algorithms. This work also details classes of evaluation metrics and ranks the MLA based on these metrics.

3. RESEARCH METHODOLOGY

3.1. The proposed architecture

In this work, we discuss the proposed architecture of the callbot system. Figure 1 shows the components of the system. The architecture includes several components: the private branch exchange (PBX) server [24], the automatic speech recognition (ASR) [25] module, the natural language processing (NLP) module, the decision-making module, and the text to speech (TTS) module.

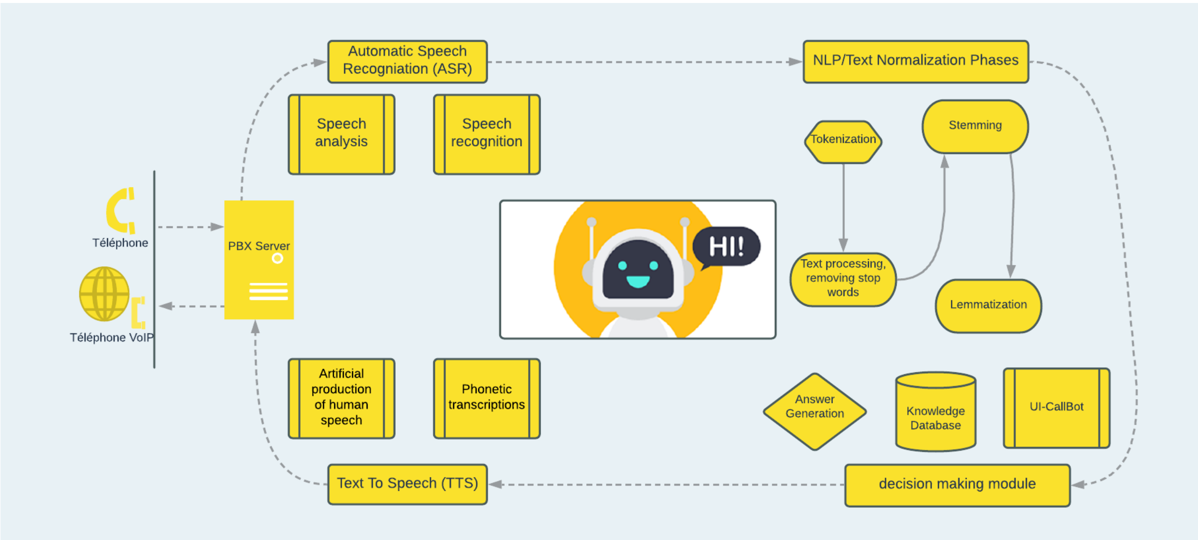


Figure 1. The proposed architecture of the AI-based enterprise callbot

The PBX server, is a private telephone network capable of handling communication between users of the telephone system within the same network or with external users, using technologies like voice over internet protocol (VoIP). The callbot process is initiated by the customer's call, this call can be made from a traditional telephone line (landline or mobile) or VoIP. The PBX server switches the call to the company's internal telephone system. The ASR handles the transcription of the client's request from their voice to text form. Then this text will be processed by the NLP module to transfer it to the decision-making part. Based on the knowledge database, the "decision-making module" generates the best(s) answer(s) before transmitting this (these) response(s) to the client. The TTS module transforms the response from the textual form into a human voice. The PBX still acts as a switch to transfer the answer to the customer.

ASR is the technique which allows a program to transform human speech from its vocal format to a text format in order to use it by the machine. it is also called the speech to text. Figure 2 shows how ASR works: the module takes the audio signal as input, the "signal processing and feature extraction" part allows signal cleaning and noise suppression to improve speech quality, the signal is then converted from the time domain to the frequency domain. the "Hypothesis Search" module combines the result of the two modules "The acoustic model" and "The language model" in order to output the sequence of words with the best result.

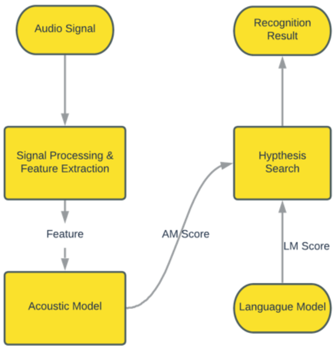


Figure 2. The architecture of ASR system [25]

3.2. NLP module

Conversational AI, such as callbots, is one of the most common contexts in which NLP [26]-[29] takes place. The purpose of this technique is to allow a fluid and natural conversation between the machine and the human, the bot integrates into a natural conversation as human as possible while moving away from static pre-senarized responses. Our callbot is designed to communicate with customers and give them the impression that they are talking to a person and not a bot. But when customers are in a conversation, they may make unnecessary sounds or may be different ways of asking the same question. Therefore, we need to preprocess the data so that its engine can easily understand it. Hence, the main task of the callbot is to be able to understand the customer's needs, called the intent. Suppose a client wishes to report a claim, the customer can express his need in several possible ways. We must identify the intent, the context and take into account everything that is discussed during the call because the client wants to get the right answer. For example, the customer may say "I want to report a water leak in my bathroom" or "I have a water damage problem". The callbot must conclude that the customer wishes to report a claim relating to his home insurance contract.

Figure 3 shows the process of the text normalization phases from the input speech to the output valid answer. The first step is the sentence segmentation, which will divide the input text into separate sentences, then the clear special characters step eliminates special characters from the text. Tokenization is the phase which consists in distilling the text into single words. These "tokens" allow the system to first identify the basic words involved in the text prior to further processing the material. Stop words are words that do not have important meanings for use in search queries. Typically, these words are excluded from search requests because they return a lot of unnecessary information. The stemming step aims to make the words in the original form in the French language avoid some problem of expressing a single word in different forms. The Lemmatization stage is the algorithmic process of determining the lemma (a canonical form) of a word depending on its expected meaning.

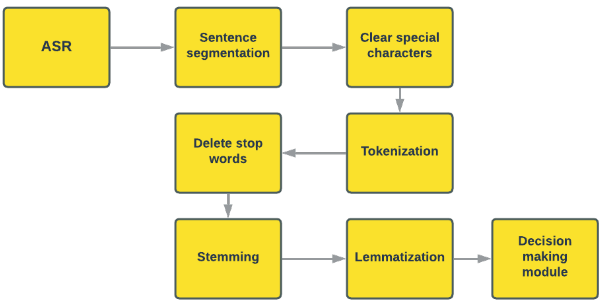


Figure 3. Flow chart of the text normalization process

3.3. Callbot user interface

The callbot's user interface (UI) allows users to develop the scenarios that the machine will use in order to meet the customer's request. it is the interface that allows to implement the knowledge base graphically without any prior technical knowledge. In Figure 4 , we have the configuration of the scenario responding to the customer request "receive a home insurance certificate". In Figure 5 we have an real client request. In addition to scenarios configuration shown in Figure 4, the interface allows: conversation monitoring, customer informations (phone number, name, contract number) as well as call recording so that the supervisor follows the operation of the bot from end to end shown in Figure 5.

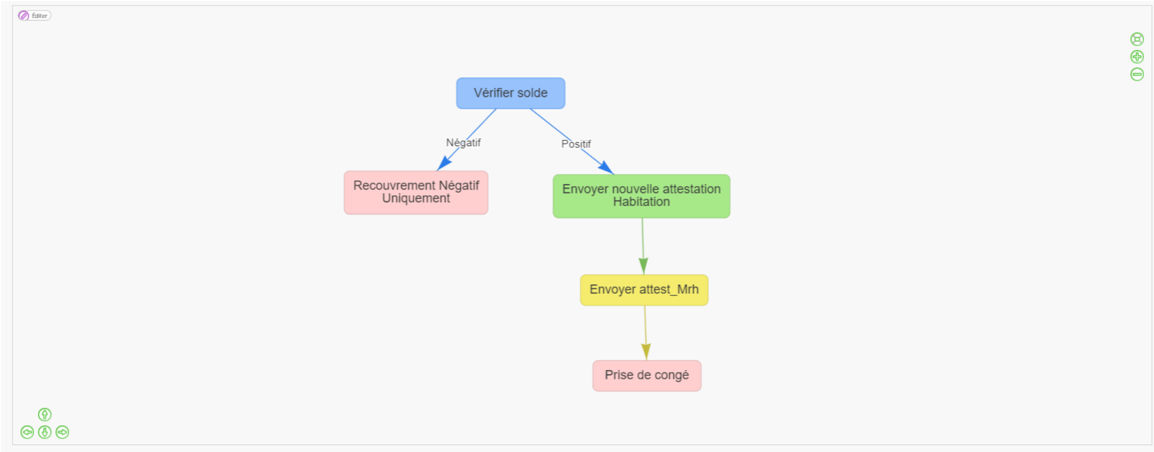


Figure 4. Example of a simple scenario

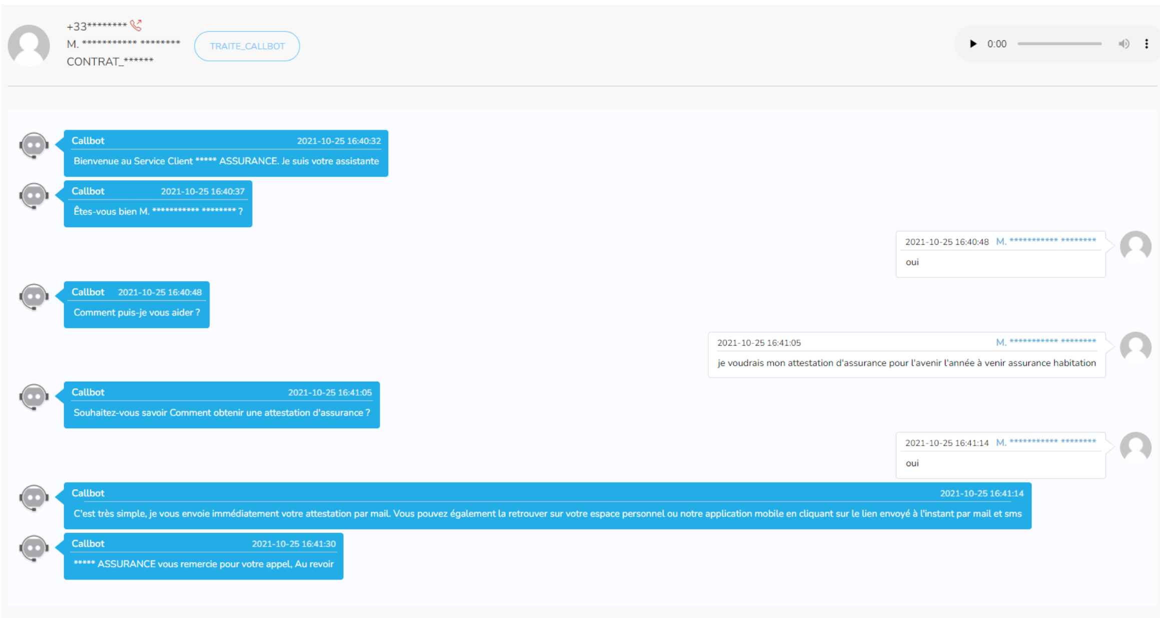


Figure 5. Illustration of conversation between the customer and the callbot

3.4. Decision-making module

This module's role is to find the answer to the customer's request. It includes the callbot-UI (CUI), the knowledge base, and the MLA. The CUI is a graphical tool developed to allow "business" users to manage the callbot system. Its main role is to create, modify or delete a scenario to feed the knowledge base of our system. This tool also makes it possible to monitor the system operations in real-time. The CUI assists users to create simple scenario capable of giving a direct answer, or complicated scenarios with several interactions. Moreover, the algorithms cited in Table 1 are the most used algorithms in ML, hence we wanted to make a comparison between all these algorithms in order to find the best one for our study. The comparison criteria considered are: the average of correct answers, the response time, the accuracy, Cohen's kappa, and F1 score.

Table 1. List of ML algorithms

Algorithm	Description	Algorithm author
SVM	The goal is to find the hyper-planes that separate the data points with the maximum margins between the two decision limits.In order to minimize generalizing errors, SVMs have the advantage of reducing the risk of exceedances.	Cortes and Vapnik [30]
RF	The Meta-estimator adapts to a number of DT classifiers on different sub-samples of the data set and uses the average to improve predictive accuracy and overflow control.	Breimen [31]
KNN	The KNN is a simple MLA. The purpose of the algorithm is to categorize objects into one of the pre-defined classes of a sample group created by ML.	Altman [32]
LR	LR is a standard probabilistic statistical classification model that has been used widely in disciplines such as computer vision, marketing, social sciences, to name a few.	Tolles and Meurer[33]
Naive Bayesian	NB is a very convenient way to learn Bayesian.It assumes that the characteristic values are conditionally independent taking into account the target value, and consequently significantly reduces the calculation cost.	Thomas Bayes (1702–61)
DT	The DT is derived from a set of labelled learning instances represented by an array of attribute values and a class label.	Belson [34]

4. RESULTS AND DISCUSSION

4.1. Algorithms comparison

This section discusses two parts, the first is the comparison between the MLA in order to select the suitable one to insert into the decision-making module. The second part presents the main statistics on the use of the proposed strategy in the call service system. In order to compare the results of the algorithms used on this article, we split our data source into two parts: a training part representing 80% of the instances used, and a fifth of the instances is used for model validation. The results are presented in the form of tables and figures in the same part.

4.1.1. Comparison using n-gram (2,2)

From the Table 2 and the Figure 6, we deduced that KNNs, Naïve Bayes, SVM and RF (50) offered an excellent response time, which does not exceed the maximum 20 ms. the rest of the algorithms take an average time varying between 30 and 60 ms. For the percentage of correct responses, the results are fair except for the KNN algorithms where the results are very poor. SVM give us the best results (precision=65.87% and accuracy=66.13%).

Table 2. Performance of algorithms using n-gram (2,2)

List of algorithms	Average correct answer (%)	Response time (s)	Accuracy (%)
SVM (svc)	65.87	0.1428	66.13
RF (200)	60.12	0.6014	56.12
RF (100)	60.18	0.3230	56.14
RF (50)	61.02	0.1842	58.12
DT	59.34	0.0909	59.30
KNN (n=5)	10.50	0.0383	16.01
KNN (n=3)	33.49	0.0382	37.32
LR	62.18	0.3194	65.14
NB	63.32	0.0387	65.88

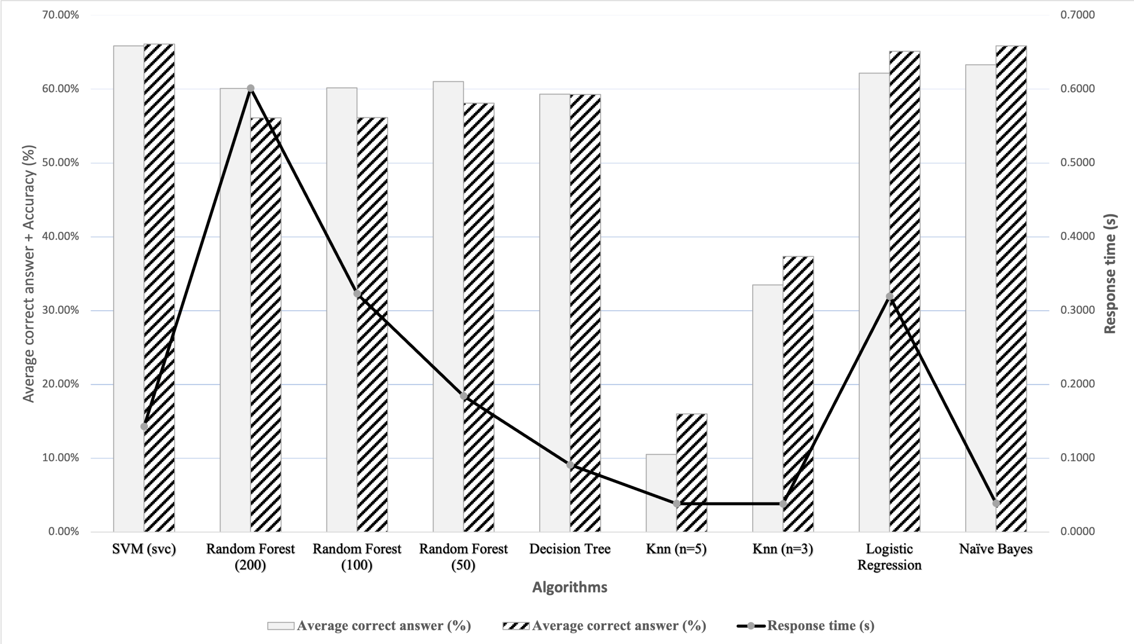


Figure 6. Performance of algorithms using n-gram (2,2)

4.1.2. Comparison using n-gram (1,1)

The results of the Table 3 and the Figure 7, show that all algorithms except RF (200 and 100) present an excellent response time, which does not exceed the maximum 25 ms. For the percentage of correct responses, there is an improvement in the overall results, but these results are still fair. Decision forests (200)/(100) and SVM give us the best results (64.18%, 61.94% and 61.94%).

In order to push the comparison further, two other comparison criteria are applied: Cohen's kappa and F1 score. As presented in Table 4 and Figure 8, for n-gram (1,1) and from the results of Cohen's kappa comparison: the strenght of agreement of the KNN (n=5) is slight (ks between 0 and 0.20), the strenght of agreement of the KNN (n=3) is poor (ks between 0.21 and 0.40), the strenght of agreement of the rest of algothims is substantial (ks between 0.61 and 0.80). For the results of Cohen's kappa for n-gram (2,2): the strenght of agreement of the KNN (n=5) is slight (ks between 0 and 0.20), the strenght of agreement of the KNN (n=3) is poor (ks between 0.21 and 0.40), the strenght of agreement of RF and DT is moderate (ks between 0.41 and 0.60), the strenght of agreement of LR, NB and SVM is substantial (ks between 0.61 and 0.80).

Based on F1 score the results of the SVM aglorithm are the best (0.62 for bigram and 0.60 from ngram (1,1)), NB and LR comes in a second ranking (F1 score between 0.56 and 0.60). KNN presents weak results (F1 score max 0.32), and the rest of the algorithms presents middling results. Using the n-gram (1,1), results of accuracy and Cohen's kappa are better, using the n-gram (2,2) average correct answer, F1 score are better. From all the results described in this section, the SVM algorithm presents the best results on all the comparison criteria.

Table 3. Performance of algorithms using n-gram (1,1)

List of algorithms	Average correct answer (%)	Response time (s)	Accuracy (%)
SVM (svc)	61.84	0.1438	70.29
RF (200)	53.24	0.6210	63.24
RF (100)	58.00	0.3254	67.65
RF (50)	53.83	0.1864	64.37
DT	54.63	0.0545	58.11
KNN (n=5)	7.17	0.0311	14.06
KNN (n=3)	15.29	0.0378	25.08
LR	58.92	0.2273	69.45
NB	58.89	0.2059	68.41

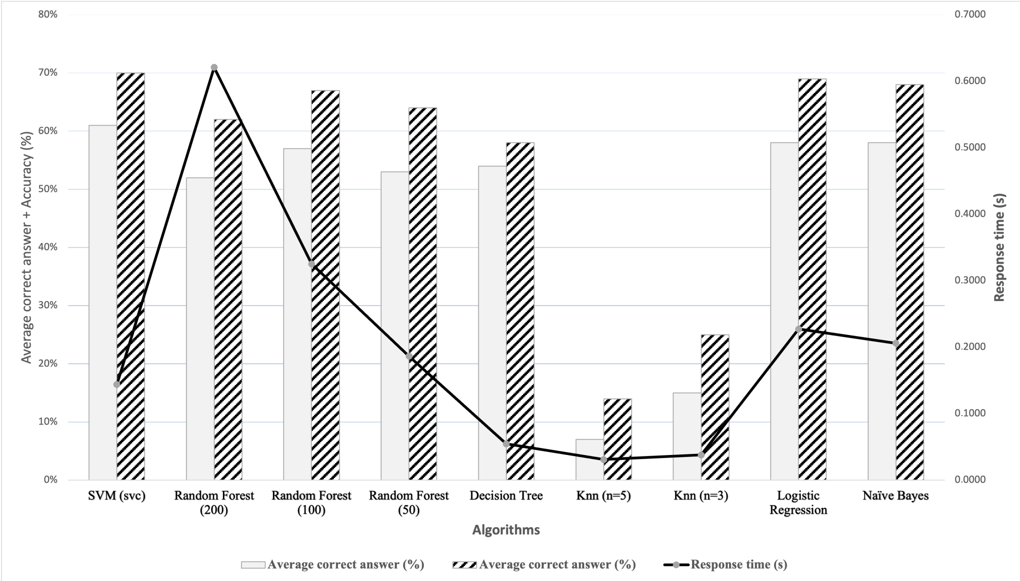


Figure 7. Performance of algorithms using n-gram (1,1)

Table 4. Results of F1 score and Cohen's kappa

List of algorithms	F1 score N-gram (2,2)	F1 score N-gram (1,1)	KS N-gram (2,2)	KS N-gram (1,1)
SVM (svc)	0.62	0.59	0.65	0.69
RF (200)	0.56	0.50	0.56	0.62
RF (100)	0.55	0.55	0.56	0.67
RF (50)	0.58	0.51	0.58	0.63
DT	0.56	0.52	0.58	0.61
KNN (n=5)	0.11	0.07	0.15	0.13
KNN (n=3)	0.32	0.16	0.37	0.25
LR	0.59	0.57	0.64	0.68
NB	0.60	0.56	0.65	0.67

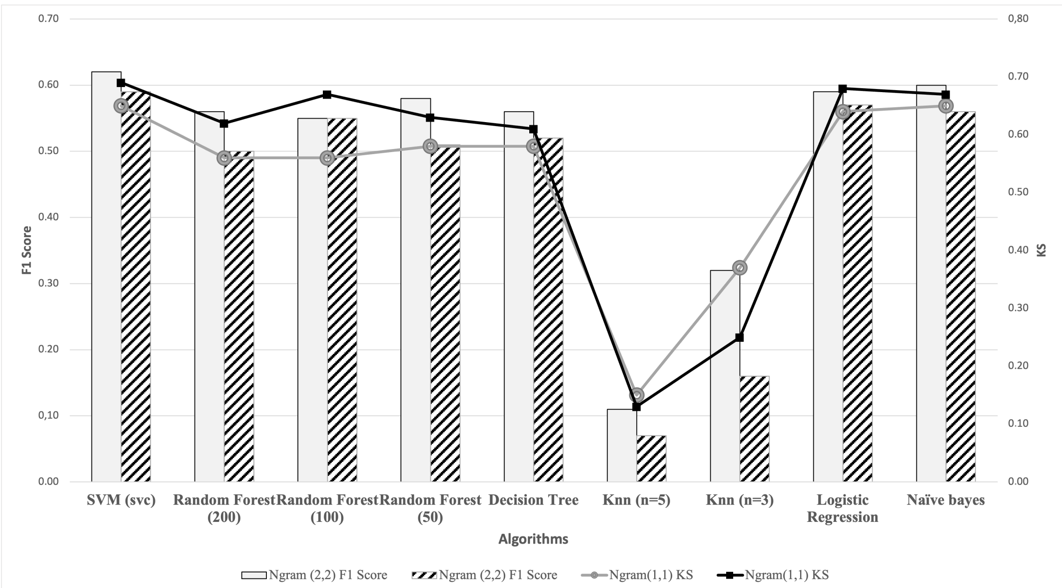


Figure 8. Presentation of results of F1 Score and Cohen's kappa

4.2. Main statics of using the callbot strategy

To test the changes noticed after the use of the callbot system in the treatment of customer calls, we first carried out statics in a period of 7 months of the year 2020 and 2021, linked to the following variables: the number of calls handled, the cost of calls, and the holding time. The statistics are collected in 4 different productions for 2 hours each. The reduced installation time is linked to the fact of not wanting to impact the quality of the service. The callbot is configured to handle 50 questions only. The rest of the requests are transferred to the agents.

4.2.1. hold time

Table 5, presents the customer waiting times before the implementation of the callbot, Table 6 and Figure 9 show the average duration that customers wait before contacting an agent. The waiting time varies from 90 seconds to more than 300 seconds. From these values, we can see that the waiting time affects the quality of the service.

Table 5. Customer waiting time before contacting an agent (without callbot)

Month	hold time (seconds)
09/2020	322
10/2020	90
11/2020	109
12/2020	145
01/2021	111
02/2021	134
03/2021	100

Table 6. Customer waiting time before processing calls when testing the callbot

	Callbot	Agents
Test N 1	0 s	156 s
Test N 2	0 s	173 s
Test N 3	0 s	40 s
Test N 4	0 s	108 s

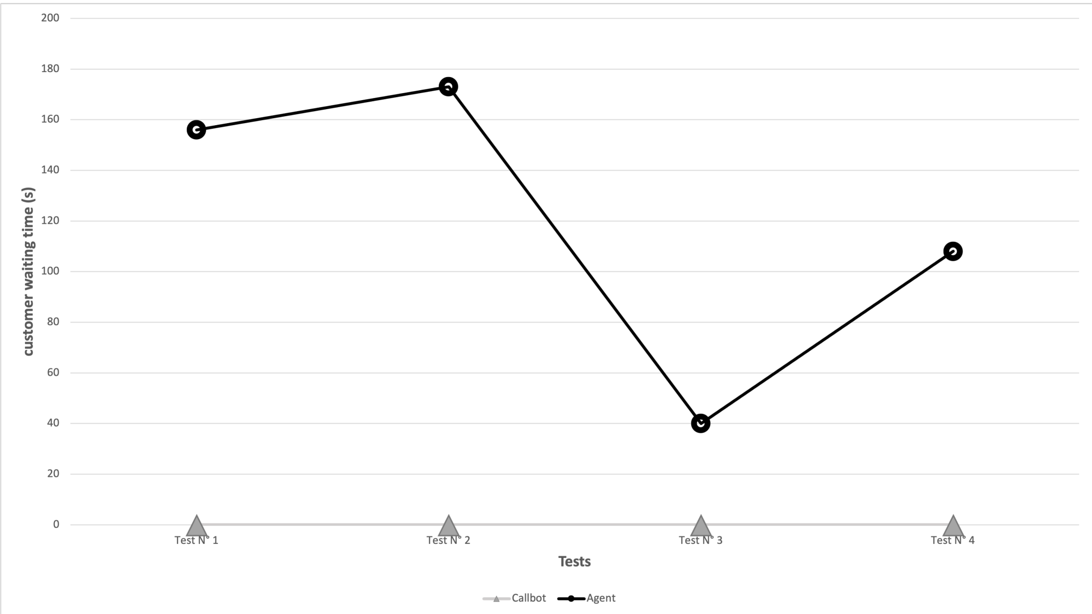


Figure 9. Customer waiting time before processing calls when testing the callbot

From Table 6 and Figure 9, we deduced clearly that the waiting time in calls handled by the callbot is very short (near 0 seconds), that is immediate processing of the customer request. Whereas the calls handled over the same period by the agents are always treated after a significant waiting time. We can also remarque that the implementation of the callbot does not necessarily improve the waiting time on calls outside the callbot perimeter.

4.2.2. Cost time and duration of the treatment

The processing time is the duration of the call between the advisor and the client. Prompt handling of the call will improve customer satisfaction. Table 7 present the changing in number of calls/costs. Table 8 compare the benifit between the cost of the calls handling by agents and calls handling by callbot. Figure 10 present the cost improvement between the callbot and calls handled by agents. And finally, Table 9 and Figure 11 present the evolution of the duration of calls handled by the callbot against the calls handled by the agents

Table 7. Example of changes in the number of calls/costs over a period of 7 months

Month	Number of calls	Cost
09/2020	35,012	140,048
10/2020	37,046	148,184
11/2020	35,430	141,720
12/2020	35,899	143,596
01/2021	40,873	163,492
02/2021	38,174	152,696
03/2021	39,149	156,596
Average	37,369	149,476

Table 8. Cost of calls when set up and the benefit compared to the agent call handling model

	calls	Cost callbot	Cost agents	Saving
Test N 1	52	23.93	208.00	88%
Test N 2	54	34.12	216.00	84%
Test N 3	53	33.14	212.00	84%
Test N 4	79	28.69	316.00	91%

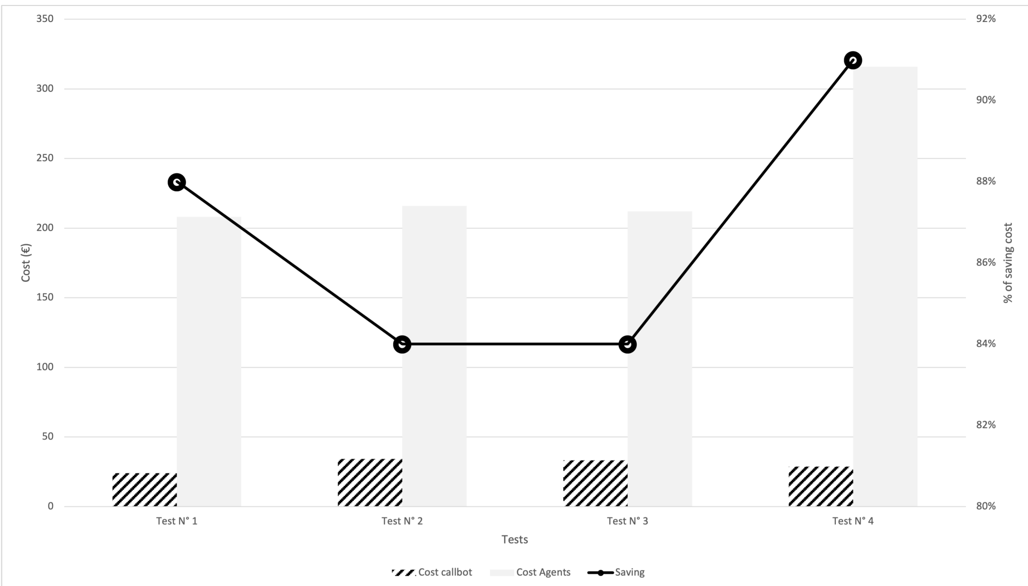


Figure 10. Presentation of cost improvement between the callbot and calls handled by agents

Table 9. Duration of calls handled by callbot vs calls handled by agents

	Callbot	Agents
Test N 1	100	261
Test N 2	80	280
Test N 3	92	259
Test N 4	112	275

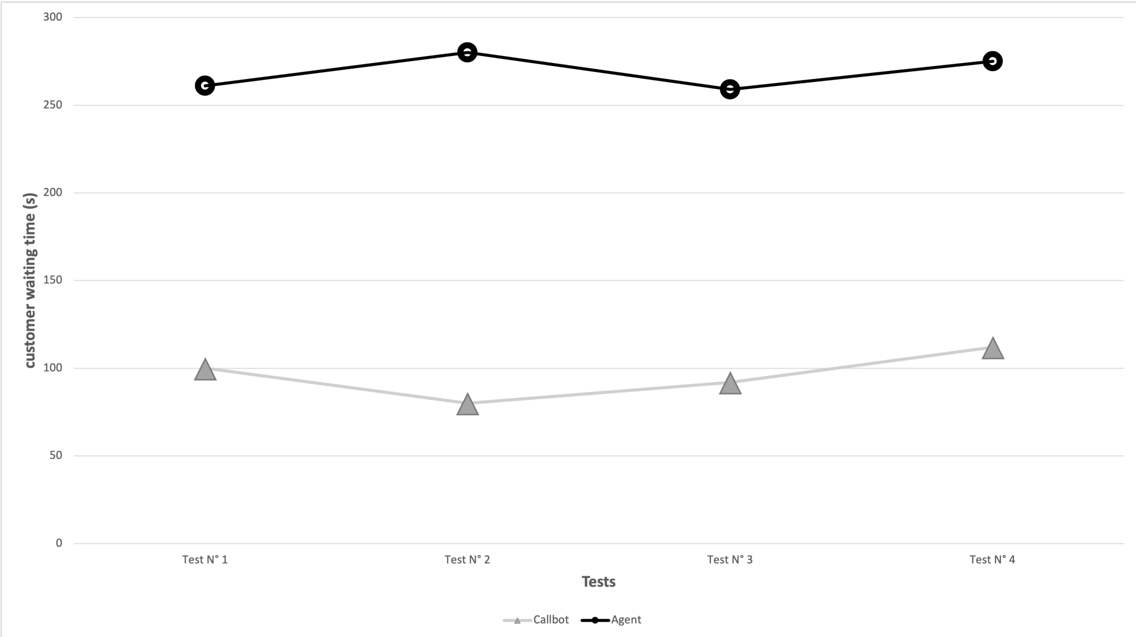


Figure 11. Duration of calls handled by callbot vs calls handled by agents

Table 7, Table 8 and Figure 10, confirmed that the cost of calls handled by the callbot is much lower than the cost of calls handled by the agents. The cost savings are greater than 80% on all the tests carried out. From Table 9 and Figure 11, we find that the processing time varies between 1 min 20 seconds and 2 minutes, while the processing time by human agent varies on average between 4 minutes 20 seconds and 4 minutes 40 seconds. The gain in processing time is very important for calls handled by the callbot, where better customer satisfaction can be expected.

5. CONCLUSION

This paper presented the architecture of our callbot system realized with NLP and ML techniques. The results obtained during this work showed that the SVM (accuracy=88.13%) and RF decision (accuracy=96.61%) algorithms are the best for implementing our decision module. The tests carried out using the proposed approach have led to very significant gains: a reduction in the cost of processing calls (the cost savings are greater than 80% on all the tests carried out), and an optimization of both the holding time and the call processing duration. Future work could include the model optimization of the knowledge base in order to increase the number of calls processed by the proposed system.

REFERENCES

[1] E. Tebenkov and I. Prokhorov, "Machine learning algorithms for teaching AI chat bots," *Procedia Computer Science*, vol. 190, pp. 735–744, 2021, doi: 10.1016/j.procs.2021.06.086.

[2] A. López, J. Sánchez-Ferreres, J. Carmona, and L. Padró, "From process models to chatbots," in *International Conference on Advanced Information Systems Engineering (CAiSE 2019): Advanced Information Systems Engineering*, 2019, vol. 11483, pp. 383–398, doi: 10.1007/978-3-030-21290-2_24.

[3] N. Teckchandani, A. Santokhee, and G. Bekaroo, "AIML and sequence-to-sequence models to build artificial intelligence chatbots: insights from a comparative analysis," in *International Conference on Emerging Trends in Electrical, Electronic and Communi-*

cations Engineering (ELECOM 2018): Smart and Sustainable Engineering for Next Generation Applications, 2019, vol. 561, pp. 323–333, doi: 10.1007/978-3-030-18240-3_30.

[4] A. Augello, M. Gentile, L. Weideveld, and F. Dignum, "A model of a social chatbot," in *Intelligent Interactive Multimedia Systems and Services 2016*, vol. 55, Springer International Publishing, 2016, pp. 637–647.

[5] O. A. Tazl and F. Wotawa, "Using model-based reasoning for enhanced chatbot communication," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2019, vol. 11606 LNAI, pp. 791–798, doi: 10.1007/978-3-030-22999-3_67.

[6] A. Kerly, P. Hall, and S. Bull, "Bringing chatbots into education: towards natural language negotiation of open learner models," *Knowledge-Based Systems*, vol. 20, no. 2, pp. 177–185, Mar. 2007, doi: 10.1016/j.knosys.2006.11.014.

[7] W. Shalaby, A. Arantes, T. G. Diaz, and C. Gupta, "Building chatbots from large scale domain-specific knowledge bases: challenges and opportunities," in *IEEE International Conference on Prognostics and Health Management (ICPHM)*, Jun. 2020, vol. 2020-June, doi: 10.1109/ICPHM49022.2020.9187036.

[8] J. Cerezo, J. Kubelka, R. Robbes, and A. Bergel, "Building an expert recommender chatbot," in *Proceedings 2019 IEEE/ACM 1st International Workshop on Bots in Software Engineering (BotSE)*, May 2019, pp. 59–63, doi:10.1109/BotSE.2019.00022.

[9] J. Lehvä, N. Mäkitalo, and T. Mikkonen, "Case Study: building a serverless messenger chatbot," in *International Conference on Web Engineering-ICWE 2017*: Current Trends in Web Engineering, 2018, vol. 10544, pp. 75–86, doi:10.1007/978-3-319-74433-9_6.

[10] R. Khan and A. Das, Build better chatbots. Berkeley, CA: Apress, 2018, doi: 10.1007/978-1-4842-3111-1.

[11] I. Aattouri, M. Rida, and H. Mouncif, "Creation of a callbot module for automatic processing of a customer service calls," in *Lecture Notes in Business Information Processing*, vol. 416 LNBIP, Springer International Publishing, 2021, pp. 417–424.

[12] I. Aattouri, M. Rida, and H. Mouncif, "A comparative study of learning algorithms on a call flow entering of a call center," in *Advances in Intelligent Systems and Computing*, vol. 1364 AISC, Springer International Publishing, 2021, pp. 507–519.

[13] B. Trstenjak, S. Mikac, and D. Donko, "KNN with TF-IDF based framework for text categorization," *Procedia Engineering*, vol. 69, pp. 1356–1364, 2014, doi: 10.1016/j.proeng.2014.03.129.

[14] K. Shah, H. Patel, D. Sanghvi, and M. Shah, "A comparative analysis of logistic regression, random forest and knn models for the text classification," *Augmented Human Research*, vol. 5, no. 1, p. 12, Dec. 2020, doi: 10.1007/s41133- 020-00032-0.

[15] M. Ikonomakis, S. Kotsiantis, and V. Tampakas, "Text classification using machine learning techniques," *WSEAS Transactions on Computers*, vol. 4, no. 8, pp. 966–974, Aug.2005.

[16] R. Ravi Kumar, M. Babu Reddy, and P. Praveen, "A review of feature subset selection on unsupervised learning," *MATTER: International Journal of Science and Technology*, vol. 28, no. 20, pp. 691–697, Feb. 2019, doi: 10.1109/AEEICB.2017.7972404.

[17] Yao, Haipeng,*et al*. "A Feature Selection Method Basedon Synonym Merging in Text Classification System". *EURASIP Journal on Wireless Communications and Networking*, vol. 2017, N 1, december 2017, p. 166. DOI.org (Crossref), https://doi.org/10.1186/s13638-017-0950-z.

[18] J. J. Khanam and S. Y. Foo, "A comparison of machine learning algorithms for diabetes prediction," *ICT Express*, vol. 7, no. 4, pp. 432–439, Dec. 2021, doi: 10.1016/j.icte.2021.02.004.

[19] M. Emami Javanmard, S. F. Ghaderi, and M. Hoseinzadeh, "Data mining with 12 machine learning algorithms for predict costs and carbon dioxide emission in integrated energy-water optimization model in buildings," *Energy Conversion and Management*, vol. 238, p. 114153, Jun. 2021, doi: 10.1016/j.enconman.2021.114153.

[20] R. Nayak, S. Amirali Jiwani, and B. Rajitha, "Spam email detection using machine learning algorithm," *Materials Today: Proceedings*, Apr. 2021, doi: 10.1016/j.matpr.2021.03.147.

[21] Neelakantan . P, "Analyzing the best machine learning algorithm for plant disease classification," *Materials Today: Proceedings*, Aug. 2021, doi: 10.1016/j.matpr.2021.07.358.

[22] D. Lv, S. Yuan, M. Li, and Y. Xiang, "An empirical study of machine learning algorithms for stock daily trading strategy," *Mathematical Problems in Engineering*, vol. 2019, pp. 1–30, Apr. 2019, doi: 10.1155/2019/7816154.

[23] I. Portugal, P. Alencar, and D. Cowan, "The use of machine learning algorithms in recommender systems: a systematic review," *Expert Systems with Applications*, vol. 97, pp. 205–227, May 2018, doi: 10.1016/j.eswa.2017.12.020.

[24] B. Jackson, C. Clark III, L. Chaffin, and J. Long, "What is asterisk and why do you need it?," in *Asterisk Hacking, Elsevier*, 2007, pp. 1–20.

[25] D. Yu and L. Deng, Automatic speech recognition. London: Springer London, p. 321, 2015, doi: 10.1007/978-1- 4471-5779-3

[26] J. Kaur and J. R. Saini, "A study of text classification natural language processing," *VNSGU Journal of Science and Technology*, vol. 4, no. 1, pp. 162–167, 2015.

[27] M. Zhou, N. Duan, S. Liu, and H. Y. Shum, "Progress in neural NLP: modeling, learning, and reasoning," *Engineering*, vol. 6, no. 3, pp. 275–290, Mar. 2020, doi: 10.1016/j.eng.2019.12.014.

[28] A. Kao and S. R. Poteet, Natural language processing and text mining. Springer London, 2007.

[29] R. V. B. Vangara, K. Thirupathur, and S. P. Vangara, "Opinion mining classification u sing naive bayes algorithm," *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, vol. 9, no. 5, pp. 495–498, Mar. 2020, doi: 10.35940/ijitee.e2402.039520.

[30] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, Sep. 1995, doi: 10.1007/bf00994018.

[31] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1023/A:1010933404324

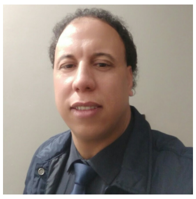
[32] N. S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," *The American Statistician*, vol. 46, no. 3, pp. 175–185, Aug. 1992, doi: 10.1080/00031305.1992.10475879.

[33] J. Tolles and W. J. Meurer, "Logistic regression: relating patient characteristics to outcomes," *JAMA - Journal of the American Medical Association*, vol. 316, no. 5, pp. 533–534, Aug. 2016, doi: 10.1001/jama.2016.7653.

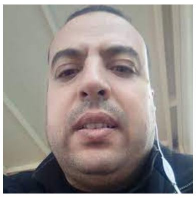
[34] W. A. Belson, "Matching and prediction on the principle of biological classification," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 8, no. 2, pp. 65–75, Jun. 1959, doi: 10.2307/2985543.

BIOGRAPHIES OF AUTHORS

Imad Aattouri received the M.Sc. degree in Computer Sciences from the Faculty polydiscilplinaire at the Sultan Moulay Slimane University, Beni-Melal, Morocco, in 2016, and now he is a Ph.D. candidate and a member of OSSI Laboratory at Ain-Chock Faculty of Sciences in the Hassan II, Casablanca, Morocco. His research interests are artificial intelligence, machine learning algorithms, and VoIP technologies. He is the corresponding author of this article. He can be contacted at email: imad.aatt@gmail.com.



Hicham Mouncif received the D.Sc. degree (Doctor Habilitatus D.Sc.) in computer science from the Faculty of Sciences and Techniques, Beni Mellal, Morocco. He has been a professor of computer science at the Ploydisciplinaire School of Beni Mellal since 2013. He is also a Director of Graduate Studies in Computer Science and Head of the Master of Informatics Systems engineering (2019-present). His research interests are in E-learning, machine learning, and GIS-technology application. He was the Head of the Computer Sciences Department at the Faculty Polydisciplinaire from 2017 to 2020. He can be contacted at email: h.mouncif@usms.ma.



Mohamed Rida D.Sc. (PhD Habilitatus D.Sc.) in Computer Science from the Ain Chock Faculty of Sciences, Casablanca, Morocco. He is a Professor of Computer Science at the Faculty of Science Ain Chock, Casablanca since 2005. Additionally, he holds the position of Director of Graduate Studies in Computer Science. His research interests are in simulation and modeling, Machine Learning, and GIS-Technology application. He can be contacted at email: mhd.rida@gmail.com.