Advising Agent for Service-Providing Live-Chat Operators

Extended Abstract

Aviram Aviv*
Bar-Ilan University
Ramat-Gan, Israel
aaviv10@gmail.com

Yaniv Oshrat*
Bar-Ilan University
Ramat-Gan, Israel
oshblo@zahav.net.il

Samuel Assefa
US Bank AI Innovation
United States
sammy.assefa@gmail.com

Toby Mustapha
J.P. Morgan AI Research
New York, United States
tmusta78@gmail.com

Daniel Borrajo
J.P. Morgan AI Research
New York, United States
dborrajo@ia.uc3m.es

Manuela Veloso
J.P. Morgan AI Research
New York, United States
manuela.veloso@jpmorgan.com

Sarit Kraus
Bar-Ilan University
Ramat-Gan, Israel, sarit@cs.biu.ac.il

ABSTRACT

Call centers, in which human operators attend clients using textual chat, are very common in modern e-commerce. Training enough skilled operators who are able to provide good service is a challenge. We propose a methodology for the development of an assisting agent that provides online advice to operators while they attend clients. The agent is easy-to-build and can be introduced to new domains without major effort in design, training and organizing structured knowledge of the professional discipline. We demonstrate the applicability of the system in an experiment that realizes its full life-cycle on a specific domain, and analyze its capabilities.

KEYWORDS

Human study; advising agent; human-agent interaction; call center

ACM Reference Format:
Aviram Aviv[1], Yaniv Oshrat[1], Samuel Assefa, Toby Mustapha, Daniel Borrajo, Manuela Veloso, and Sarit Kraus. 2022. Advising Agent for Service-Providing Live-Chat Operators: Extended Abstract. In *Proc. of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022), Online, May 9–13, 2022*, IFAAMAS, 3 pages.

1 INTRODUCTION

In modern e-commerce, many business services are provided via the Internet. For example, banks are increasingly closing their physical branches and moving services, formerly provided only face-to-face, to the internet [7]. There are many actions that customers can perform by themselves, without human intervention, either by a self-service application or by maintaining a textual chat with a conversational chatbot. However, when customers want to perform actions that do not yet have an online solution, or when they fail to do it by themselves, they still need to approach the bank's customer service and seek human help. In these cases, human operators attend to the customers via the textual chat channel. A single human operator may attend to several customers simultaneously.

While this approach has its advantages, it also raises some challenges for the human operator to deal with. As the number of tasks that the operators have to perform simultaneously grows, so may their stress. Operators also need to prioritize the tasks, keep track of each individual's information while assisting different clients, and provide help without making any client wait too long. We propose to mitigate these challenges by assisting the human operator in creating an **advising agent** [8]. This kind of agent works alongside the operator during the chat session, and suggests on-line advice to help the operator deal with a given situation. However, building an advising agent and training it to the specific service domain can be a long and expensive process that requires both domain expertise and system engineering knowledge.

In this work we present a design for an automated agent that assists the operator during textual chat interactions with customers in real-time, by providing the operator with advice about possible actions and relevant information. Our design combines standard ML methods with domain-expert annotations, and tries to predict the actions and suggestions of the expert. The novelty of our method is twofold: First, the assistance of the agent is not focused on providing answers to customers' questions (as in former works, e.g [2, 3]), but rather in guiding the operators as to what questions they should ask in order to get the required information to provide service. Second, the process of training the agent to a new domain is short and does not require many resources or domain knowledge from outer sources. Finally, we field-test our design on a specific domain and present our findings.

2 THE PROPOSED MODEL

2.1 Agent's Life-Cycle

The process of building an advising-agent for a new domain is performed in three phases, as follows:

(1) **The Apprentice Phase (Phase 1)** – experienced human operators serve human customers regarding the new domain of service. The operators tag the information they find important in the chat conversation: They may do it in real-time,

*Equal contribution

Proc. of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022), P. Faliszewski, V. Mascardi, C. Pelachaud, M.E. Taylor (eds.), May 9–13, 2022, Online.

as the chat goes on, or afterwards. The collected data is fed to the learning process. This phase exists only for the sake of collecting information for the next phases, and does not include any agent assistance.

(2) **The Novice-Advisor Phase (Phase 2)** – this phase contains both data collection and service to clients: the agent works alongside a non-experienced human operator who attends clients, and it simultaneously advises and learns. For advising the human operator, the agent uses the tagging from the chat conversation to predict messages that the operator should send or actions it should perform, and offers them to the operator. The operators may use this advice or not, as suits them. In addition, the data collected in this phase may be fed into the agent's machine learning model in order to improve its tagging and advising capabilities.

(3) **The Expert-Advisor Phase (Phase 3)** – The agent works alongside a non-experienced human operator and provides advice based on former tags and a learned model. The agent is not engaged in further learning, since its capabilities have already reached an adequate level. This phase is the final and steady state of the agent in the current domain.

2.2 The Learning Process

In order to provide advice, the agent relies on a predictive model learned from observations of the domain: Operators conduct chat sessions with clients and attend to their needs. During the chat sessions, the operators tag the vital information items they used to reach the satisfactory outcomes. An information item may include a single word or a phrase, and it depends on the specific domain in which the service is provided. All the tagging is done during the chat conversation or after it; there is no tagging in advance.

Each session's tag-list is turned into an **information vector**. Each time a new tag is added, the vector's current version is saved to be used later in the learning process as an information vector.

For the learning algorithm, we wanted to find an algorithm with the ability to work efficiently on several domains and handle messy and conflicting data. The first model that came to mind was Random Forest [1], a model that works well but cannot fully utilize the vast amount of data usually available in such domains. To deal with this problem, we thought of using neural networks. That idea was relatively successful, but an architecture that works on one domain might fail to learn on another. With all that in mind, we decided to combine them as an ensemble method of neural networks [4] where each network takes the information known about a customer at a certain time and outputs the recommended set of advice for the situation. Each network in the ensemble was trained on a subset of the data and had a random number of layers of an arbitrary length.

The final set of advice was chosen using a majority voting variation. We also tested this method against other variations of Random Forest (LGBM [6] and regular Random Forest) and other crowd related algorithms (SVM and KNN). This method outperformed the others in an 80:20 cross-validation where the target label needed to be in the top 2 recommendations (the ensemble reached 87% accuracy, regular and gradient boosted Random Forests with 84%, KNN with 83%, neural network with 77% and SVM with 70%). We chose this metric because there can be a large variation based on the operator's preferences, even with a small amount of data.

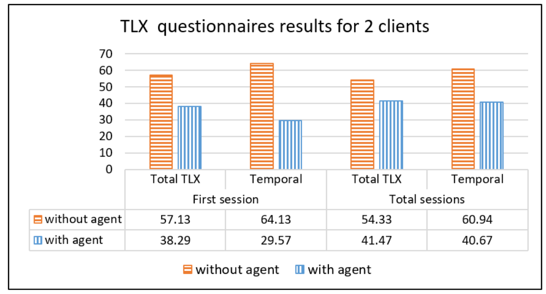

Figure 1: NASA-TLX questionnaires' data (lower is better).

3 EXPERIMENT

We performed a user experiment with human subject to test our model. We chose the domain of students loans in the US, and implemented a "WhatsApp-like" chat interface as a textual channel. The experiment articulates Phases 1 and 2 in the aforementioned model: In the first part of the experiment, testing Phase 1, four subjects played the role of a human operator in 76 sessions, and the collected data was used to train the virtual agent. In the second part, testing Phase 2, 29 subjects played the role of the operator, each of them playing two sessions: One with an assisting agents and one without it. At the end of each session, we asked the participants who played the operators to fill out a NASA-TLX questionnaire [5]. These opinions were analyzed in order to evaluate the performance of the agent and its contribution to the performance of the operators.

We measured the performance of the operators in two methods: The subjective grades the participants filled out in the NASA-TLX questionnaires, and the objective time performance of the service they provided. The questionnaires analysis showed that in the total TLX and in the temporal TLX, the operators that were assisted by a virtual agent experienced less workload than those who were not assisted. In all the time performance measures the service of the assisted operators was 2.5% to 31% shorter than the non-assisted.

4 CONCLUSIONS

We suggested a three-phase process to build and to train an assisting agent for textual service to clients. The experiment we performed to test this process showed that it is possible to train this model using on-going service sessions of real human operators, without any preliminary domain information, and the agent built using this process can improve the performance of human operators.

ACKNOWLEDGMENTS

Disclaimer: This paper was prepared for informational purposes by the Artificial Intelligence Research group of J.P. Morgan Chase & Co. and its affiliates ("J.P. Morgan"), and is not a product of the Research Department of J.P. Morgan. J.P. Morgan makes no representation and warranty whatsoever and disclaims all liability, for the completeness, accuracy or reliability of the information contained herein. This document is not intended as investment research or investment advice, or a recommendation, offer or solicitation for the purchase or sale of any security, financial instrument, financial product or service, or to be used in any way for evaluating the merits of participating in any transaction, and shall not constitute a solicitation under any jurisdiction or to any person, if such solicitation under such jurisdiction or to such person would be unlawful.

REFERENCES

[1] Leo Breiman. 2001. Random forests. *Machine learning* 45, 1 (2001), 5–32.
[2] William Cheetham. 2003. Lessons Learned using CBR for Customer Support.. In *FLAIRS Conference*. 114–118.
[3] Roland Graef, Mathias Klier, Kilian Kluge, and Jan Felix Zolitschka. 2020. Human-machine collaboration in online customer service–a long-term feedback-based approach. *Electronic Markets* (2020), 1–23.
[4] Lars Kai Hansen and Peter Salamon. 1990. Neural network ensembles. *IEEE transactions on pattern analysis and machine intelligence* 12, 10 (1990), 993–1001.
[5] Sandra G Hart, Lowell E Staveland, et al. 1988. Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. 52 (1988), 139–183.
[6] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in neural information processing systems*. 3146–3154.
[7] Kiheung Nam, Zoonky Lee, and Bong Gyou Lee. 2016. How Internet has reshaped the user experience of banking service? *KSII Transactions on Internet & Information Systems* 10, 2 (2016).
[8] Ariel Rosenfeld, Noa Agmon, Oleg Maksimov, and Sarit Kraus. 2017. Intelligent agent supporting human–multi-robot team collaboration. *Artificial Intelligence* 252 (2017), 211–231.