Black-Box Testing of Financial Virtual Assistants

Iosif Itkin[1], Elena Treshcheva[1], Luba Konnova[1], Pavel Braslavski[2], Rostislav Yavorskiy[3]
[1]Exactpro Systems, London, UK
[2]Ural Federal University, Yekaterinburg, Russia
[3]Surgut State University, Surgut, Russia
{iosif.itkin,elena.treshcheva,luba.konnova}@exactpro.com, pbras@yandex.ru, javorski_re@surgu.ru

***Abstract*—We propose a hybrid technique of black-box testing of virtual assistants (VAs) in the financial sector. The specifics of the highly regulated industry imposes numerous limitations on the testing process: GDPR and other data protection requirements, the absence of interaction logs with real users, restricted access to internal data, etc. These limitations also decrease the applicability of a few VA testing methods that are widely described in the research literature. The approach suggested in this paper consists of semi-controlled interaction logging from the trained testers and subsequent augmenting the collected data for automated testing.**

***Index Terms*—intelligent virtual assistant, financial software**

I. Introduction

Virtual assistants (VAs) are intelligent software agents, which provide conversational output in response and can execute certain tasks. Over the latest decade, a number of banks have launched their VA services. A recent survey on machine learning in the UK financial services [1] identifies VAs as a primary example of AI applications in finance, and specifically in the customer engagement domain.

Unlike traditional financial software, the VA interacts with the user through a dialogue in natural language, while applying third-party services to obtain information and perform various actions. In technical terms, the VA collects heterogeneous data (client requests, personal information, etc.) and uses machine-learning algorithms to analyse it and enhance the quality and the individualization of VA's reactions.

However, the use of VAs in banking is associated with economical and technical risks, as shown by Budzinski et al. [2]. One of the illustrations is a technologically advanced banking app: having been trained on the open-source text data, a chatbot reacted to a client's enquiry about fingerprint login with a response "You'd better had your fingers cut," which caused reputational losses for the bank, despite an apparent human-like nature of the phrase.

The evaluation of conversational systems is a challenging task due to the high variability of user inputs. Despite many existing efforts in unsupervised evaluation of task-oriented systems [3], Liu et al. [4] show that the automatic metrics correlate poorly with human judgements. Thus, human evaluation is still needed for task-oriented systems, especially in fault-intolerant domains such as finance.

II. Approach

Testing financial VAs is even more compounded by data sensitivity, restricting a testing team's access to the existing user/chatbot interaction records. In this section, we outline our hybrid two-stage approach to overcome this data deficiency issue: first, we collect logs in a semi-controlled manner; then, we leverage the collected data for automated tests using NLP techniques as well as additional data sources. We assume that a virtual assistant to be tested is a task-oriented dialogue system in a financial domain that offers a range of skills. The skills are known and sufficiently described in advance. Moreover, the assistant permits a mixed initiative, i.e. the conversation is not steered solely by the system, but the user can also be proactive – e.g. by switching a topic/target skill or asking clarifying questions. We assume that we have access to a text API endpoint of the VA that provides an interface to a realistic test environment.[1] In addition, we have access to the internal states of the test environment, including states and actions resulting from a dialogue with the user. Another assumption is the ability to reset the system to the initial state. At the same time, we presume not having access to the code, algorithms, internal interpretation of the users' utterances, data that has been used for training the system, or to historical logs of interaction with real users. This allows us to consider our approach as a variant of black box testing.

In the first phase, we aim to collect interaction logs in semi-controlled settings.[2] For each assistant's skill, we elaborate a set of task descriptions with background information and parameter values. For example, for a money transfer task, the description contains the customer's name, credentials, the sender's and beneficiary's bank details, amount, currency, and other necessary data. After that, testers get a suite of tasks they have to accomplish using the assistant (the same task can be assigned to several testers), see Figure 1. The result is a collection of interaction logs for each of the VA's skills. We estimate the target volume of logs obtained this way at several thousands for each skill that can be collected within a couple of days (10–20 tasks performed by 100–200 testers, each task taking several minutes). Despite a relatively small volume, the logs can provide reliable information, e.g. the share of successfully completed tasks and the average time/number of turns to complete the task.

In the second phase, we annotate log fragments according to the predefined set of the VA's skills: as being greetings,

[1]We don't consider a case of voice VAs, but the principles described below can be applied to systems based on speech interfaces as well.

[2]It is possible to conduct initial experiments with existing open datasets such as MetaLWOz [5] if they cover tasks and skills of the VA to be tested.

978-1-7281-8915-4/20/$31.00 ©2020 IEEE
DOI 10.1109/QRS-C51114.2020.00120

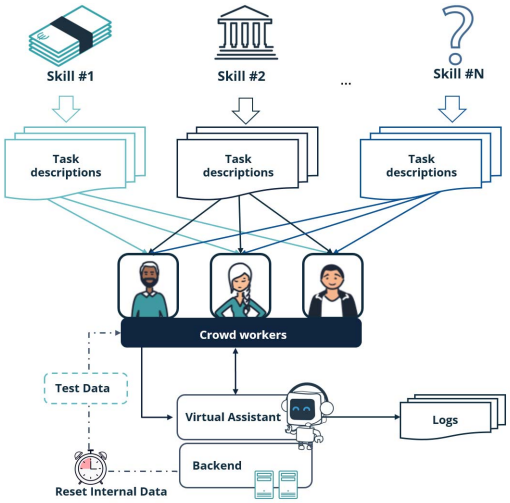

Fig. 1. Log collection.

authorization, intent indication, information input, assistant's clarifying questions, user's questions, user's dissatisfaction signals, etc. Based on the annotation, we can identify the problems of specific skills, e.g. based on the number of clarifying questions and dissatisfaction signals. More importantly, the annotation provides insights into the VA's dialog states and scenarios behind each skill. On top of formalising the scenarios, we can augment the obtained data for subsequent automated testing aimed at the assessment of the VA's overall robustness. For example, we have such a log extract (U stands for user Jack, A – for the virtual assistant):

A: *Hello, Jack. How can I help you?*
U: *I've made a transfer from another bank but can't see the money on my account.*

We can gradually modify the user's utterances to explore the tolerance of the VA to different kinds of noise in the input data. For that, valid user utterances obtained from the interaction logs are to be subjected to NLP-leveraged transformations, e.g. we can automatically introduce misspellings (1), word order violations based on syntactic parsing (2), substitute words with their synonyms or hypernyms using thesauri (3), and paraphrase the user's utterances (4):

(1) *I've made a transffer from another bank but can't see the many on my acount.*
(2) *from another bank a transfer I've made but the money on my account can't see .*
(3) *I've made a transfer from another financial institution but can't see the funds on my account.*
(4) *I've transferred money from my account at another bank but it doesn't seem to have arrived here yet.*

As long as we believe we retain the meaning of the initial expression, we expect the system's response to be identical to the original version. This approach is close to generation of adversarial examples applied to question answering and sentiment analysis proposed by Ribeiro et al. [6]. Note, however, that their approach is applied to tasks with already existing test data with unambiguous ground-truth answers.

Another possible approach to leverage the collected logs is to test user intent detection by "mixing" the phrases that signal the user's intent (as per the annotation mentioned above) from scenario templates associated with different VA skills. For example, we can automatically reproduce a dialog from the log to a certain point, then generate a response from another scenario (marked with *):

U: *Hi, I would like to withdraw money from Platform X.*
A: *Sure, what it the amount of the withdrawal?*
U: **I've made a transfer from another bank but can't see the money on my account.*

The expected system's response would be a prompt for switching to another intent or a clarifying question. To stress-test the VA's ability to resist the input noise, the approach can be extended using a large collection of realistic, but irrelevant utterances, e.g. from a corpus of movie subtitles.[3]

III. Conclusion

This short paper addresses the problem of testing VAs in the domain which is extremely sensitive to data privacy and imposes limitations on test data availability. The proposed approach is based on collecting logs obtained from the interactions between manual testers and the VA under test, and annotating them in accordance with a set of the VA's skills. It allows to subsequently use this data to generate test scenarios for testing the VA's performance on different levels – from the ability to process textual input (with spelling, lexical, syntactic variations) to the ability of identifying the user's intent / VA's skill area and responding with an appropriate utterance and/or action. The approach is focused on such quality attributes of conversational agents as performance, functionality, accessibility, as per [7]. Future research is to address other characteristics contributing to the financial VA's effectiveness, efficiency and overall user satisfaction.

References

[1] C. Jung *et al.*, "Machine learning in UK financial services," Bank of England, Tech. Rep., October 2019.
[2] O. Budzinski, V. Noskova, and X. Zhang, "The brave new world of digital personal assistants: Benefits and challenges from an economic perspective," *Ilmenau Economics Discussion Papers*, no. 118, 2018.
[3] J. Gao, M. Galley, and L. Li, "Neural approaches to conversational AI," *Foundations and Trends in Information Retrieval*, vol. 13, no. 2-3, pp. 127–298, 2019.
[4] C.-W. Liu, R. Lowe, I. V. Serban, M. Noseworthy, L. Charlin, and J. Pineau, "How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation," in *EMNLP*, 2016, pp. 2122–2132.
[5] S. Kim, M. Galley, C. Gunasekara, S. Lee, A. Atkinson, B. Peng, H. Schulz, J. Gao, J. Li, M. Adada *et al.*, "The eighth dialog system technology challenge," *arXiv preprint arXiv:1911.06394*, 2019.
[6] M. T. Ribeiro, S. Singh, and C. Guestrin, "Semantically equivalent adversarial rules for debugging NLP models," in *ACL*, 2018, pp. 856–865.
[7] N. M. Radziwill and M. C. Benton, "Evaluating quality of chatbots and intelligent conversational agents," *arXiv preprint arXiv:1704.04579*, 2017.

[3]https://www.opensubtitles.org/