August 14, 2019

# Expediting and Enhancing User Interaction with Interactive Voice Response Systems Utilizing Machine Learning

Siddartha Janga

# Expediting and Enhancing User Interaction with Interactive Voice Response Systems Utilizing Machine Learning

**Abstract:**

This publication describes a technique and methods directed at expediting and enhancing human interaction with interactive voice response (IVR) systems. The technique incorporates the utilization of a machine-learned (ML) system which can interact (*e.g.*, receive and respond to questions) with IVR systems. After sufficient interaction an IVR system, the ML system can generate an application. The application can provide a user interface (UI) which is displayable on a computing device (*e.g.*, cellphone, tablet, watch) for a user. Through the UI, the application can present the questions that the IVR system asks the user. Upon receiving user input, the ML system can communicate the user input to the IVR system.

**Keywords:**

Machine-learning (ML), interactive voice response (IVR) systems, computing devices, telecommunication services, software application, digital distribution service, customer service, DTMF tones

**Background:**

Users of computing devices (*e.g.*, cellphones, tablets, watches) often encounter interactive voice response (IVR) systems when making telephone calls. IVR systems are automated telecommunication services that guide users through a series of questions, retrieve user input, and

perform appropriate actions in response to the user input. IVR systems are broadly employed, such as in the sales, marketing, and medical fields.

The following example illustrates a common interaction between a computing device user (John) and an IVR system. John uses his cellphone to call a local restaurant to verify his dining reservation. The restaurant's IVR system accepts the call and asks John a series of questions in order to determine his desired action(s) or question(s). John, in response to the questions asked by the IVR system, presses numbers on the keypad of his phone (*e.g.*, virtual numbers displayed on the user interface of his cellphone). For instance, in the first series of questions the IVR system asks John to press the number one if he would like to interact using the English language, and in response John presses the number one. After the IVR system collects the necessary user input, the IVR system can then provide John the information he desires.

IVR systems permit corporations (*e.g.*, businesses, companies, government entities) to assist many callers in an efficient and effective manner. In order to further increase the usefulness of IVR systems, a machine-learned system can generate an application by which users may more conveniently and efficiently respond to IVR systems.

**Description:**

This publication describes a technique and methods aimed at expediting and enhancing human interaction with interactive voice response (IVR) systems. The technique incorporates the utilization of a machine-learned (ML) system that interacts with IVR systems to generate an application.

The ML system is iteratively trained to interact with IVR systems, such that the ML system can receive and respond to questions asked by an IVR system. The responses (*e.g.*, dual-tone

multi-frequency signaling (DTMF) tones associated with specific numbers, an automated voice) provided by the ML system are contextually determined. In addition, the ML system is trained to generate an application. Once training of the ML system is complete, it can be utilized to generate applications for corporations (*e.g.*, businesses, companies, government entities).

Figure 1 illustrates an ML system interacting with an IVR system to generate an application.

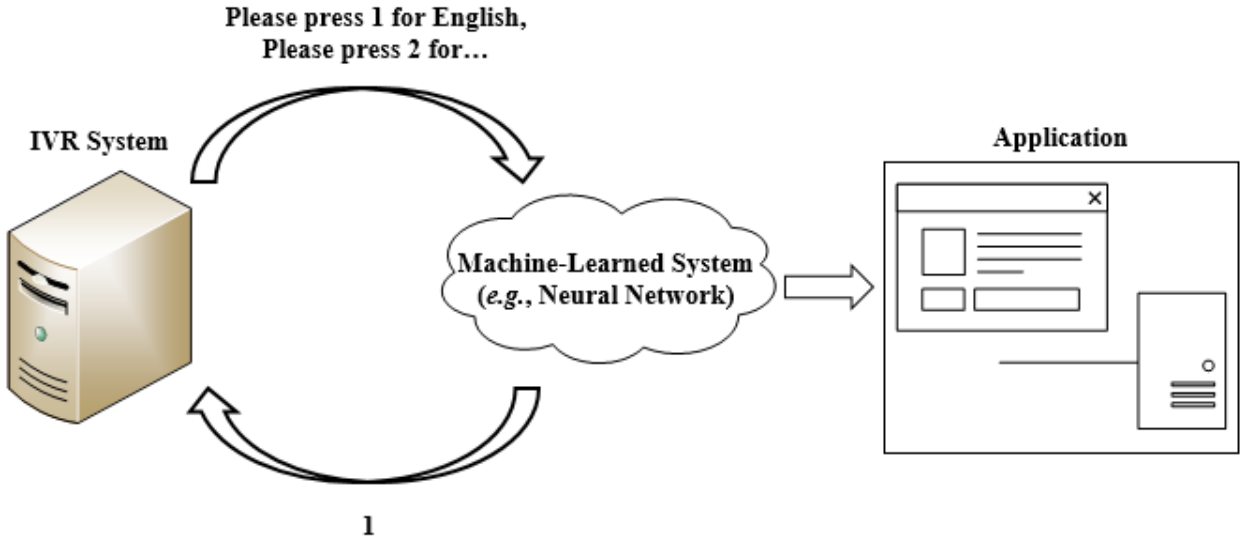


**Figure 1**

As illustrated in Figure 1, the ML system interacts with an IVR system through a telephone call. For instance, the IVR system may ask the ML system as to the preferred language ("Please press 1 for English, please press 2 for…") and the ML system can respond by selecting the number one. Through multiple iterations, the ML system can respond to all the different available options presented by the IVR system. In other words, the ML system can crawl through the IVR system. After sufficient interaction with the IVR system, the ML system can generate an application on a server. The application can then be uploaded to a digital distribution service. Computing devices can, subsequently, download an application specific to the IVR system they desire to interact with.

Figure 2 illustrates an example computing device and elements of the computing device that support the methods of user interaction with an IVR system afforded by the application.

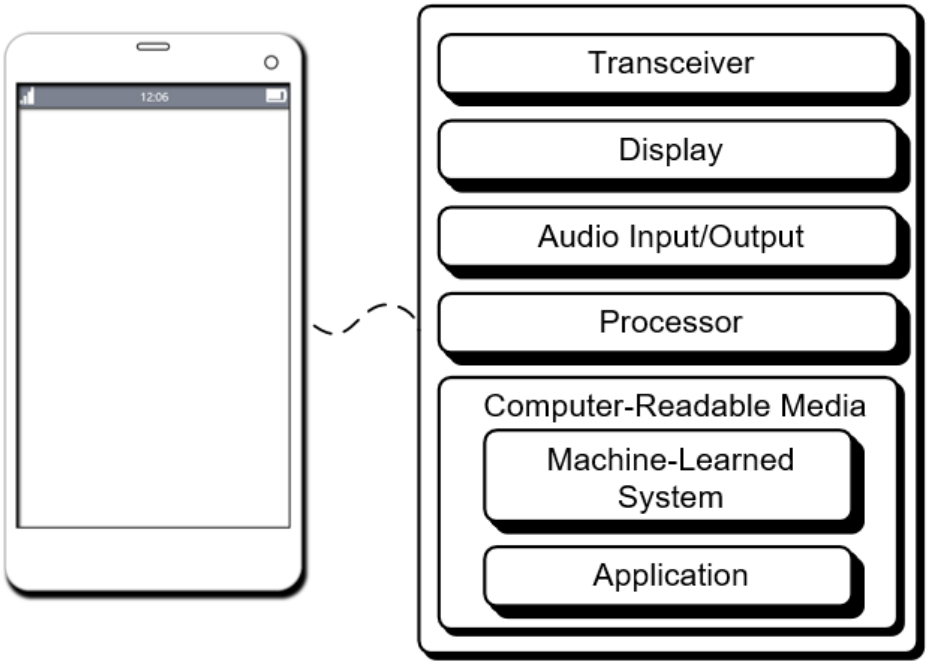


**Figure 2**

As illustrated, the computing device is a smartphone. However, other computing devices (*e.g.*, wireless-communication devices, tablets, watches, laptops) can also support the methods described in this publication. The computing device includes a transceiver(s) for transmitting data to, and receiving data from, the access point of the wireless network, a display, and an audio input/output mechanism (*e.g.*, a microphone, a speaker). The computing device also includes a processor and a computer-readable medium (CRM) that may include the operating system (OS) of the computing device and the ML-generated application for a given IVR system.

The application can provide a user interface (UI) which contains virtual buttons and fields for accepting textual input from a user. The UI is constructed based on the questions that the IVR system asks a computing device (*e.g.*, wireless-communication device, tablet, watch) user.

Figure 3 illustrates the UI provided by the application.

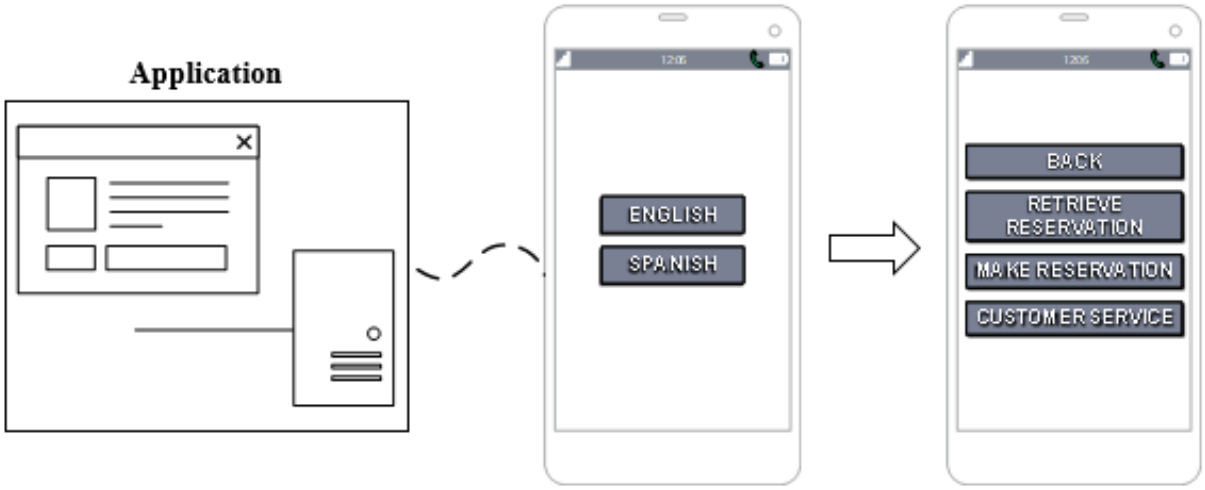


**Figure 3**

As illustrated in Figure 3, the computing device of Figure 2 displays the UI of the application. The series of questions that the IVR system would ask a caller are presented as selectable buttons on the display of the computing device. As illustrated, the first series of questions asks whether the user would like to proceed using the English or Spanish language. Upon selection, a back button along with the next series of questions ("Retrieve Reservation, Make Reservation, Customer Service") can be presented.

Once the user has addressed all the series of questions, the UI can present a submit button. Upon the user selecting submit, the application can direct a virtual assistant on the computing device or on a remote server to make a call on behalf of the user and provide the user-input at the appropriate time to the IVR system using DTMF or other forms of communication.

Further to the descriptions above, the UI may present a text field for the user to type a response into depending on the nature of the questions asked by the IVR system. For instance, if the IVR system asks for the reservation number, then the UI can display the text, "Reservation Number:" and a text field for the user to input the number.

Furthermore, as users interact with IVR systems through the applications generated by the ML system, the virtual assistant can keep track of the popular services. With this feature, the virtual assistant can update the application and provide the popular service options on the initial screen of the UI. For example, a computing device user (John) desires to verify his dining reservation at a local restaurant. Instead of calling, he navigates on his computing device to the restaurant's application installed on his computing device. On the initial screen of the application's UI, John may be presented popular service options and John quickly finds the Check Reservation Status option. Upon selecting this option, a text field may be presented asking John for his reservation number. Once John is finished inputting the number and clicking submit, the virtual assistant on a remote server can call the IVR system, provide the user-input, retrieve a response, and send the output from the IVR system to the user.

Alternative to the previously described methods, the ML system does not need to comprehensively crawl through the IVR system to generate the UI of an application. Instead, the ML system can develop the UI of the application in real-time as well.

In addition to the above descriptions, a user may be provided with controls allowing the user to make an election as to both if and when systems, applications, and/or features described herein may enable collection of user information (*e.g.*, information about a user's social network, social actions, social activities, profession, a user's preferences, a user's current location), and if the user is sent content and/or communications from a server. In addition, certain data may be treated in one or more ways before it is stored and/or used, so that personally identifiable information is removed. For example, a user's identity may be treated so that no personally identifiable information can be determined for the user. In another example, a user's geographic location may be generalized where location information is obtained (such as to a city, ZIP code,

or state level), so that a particular location of a user cannot be determined. Thus, the user may have control over what information is collected about the user, how that information is used, and what information is provided to the user.

Additional benefits of the technique and methods disclosed herein include assisting users who have difficultly hearing and/or speaking. For such users, listening and/or responding to IVR systems may be challenging. Thus, utilizing a ML system to present the IVR system's questions in a digital form through an application can complement IVR systems and aid users. Furthermore, the application provides users the opportunity to go through the questions asked by the IVR system at their preferred pace. Finally, businesses, companies, *etc.* do not need to manually create or maintain software applications dedicated for providing similar IVR system services because the ML system can generate and maintain these applications.