WHITEPAPER

PERSISTENT

Powering voice agents using IBM Watson

A White Paper
30-Mar-2017

Jaydeep Ayachit, Persistent Systems Ltd
Manish Raj, Persistent Systems Ltd

Abstract

Customer relationship and support is key for companies and enterprises to succeed. Companies spend hours and hours training resources to handle and improve customer support and interaction. This however is muted to certain extent by training cost, lack of extensive domain knowledge and business process, outsourcing challenges and infrastructure challenges.

This white paper discusses techniques and options to handle and improve customer support and interaction through use of voice agents. Customer service agents does this job with the help of IVR systems. This many times results into higher waiting time for customers, poor customer experience and call switching. Intelligent voice agents powered by natural language understanding and conversation capabilities can help alleviate the pain points and result in better customer service.

After reading through this white paper, you will be able to understand and identify strategy for deploying voice agents in your infrastructure. Persistent Systems Ltd can work with you to help build, train and deploy intelligent voice agents for your customer service.

Contents

1. Introduction

Customer relationship and support is key for companies and enterprises to succeed. Companies spend hours and hours training resources to handle and improve customer support and interaction. This however is muted to certain extent by training cost, lack of extensive domain knowledge and business process, outsourcing challenges and infrastructure challenges.

Having elaborate FAQs, user manual on websites is no longer amuse customers. People prefer phone calls to reading long FAQ or interacting over web. Companies and enterprises are adopting to innovative ways to interact with customer and improve relationship and service. “Chatbots” have surged in recent past that assist users in common day to day activities ranging from getting information, booking a ticket to placing food order. More and more providers are making available cognitive services powered by machine learning to reduce gap and improve human-machine- interaction. More and more questions get asked to Siri, Google Now and Cortana across countries, multiple languages and regional dialects.

Voice and natural language is the new way of interacting with machines. Using voice and natural language understanding makes it possible to reach to a larger audience, simplify operations and build a more lasting relationship with users and customers.

This white paper discuss and presents various options as how traditional IVR (Interactive Voice Response) systems can be made smarter to make use of cognitive services, natural language understanding and natural language processing. Typical use of IVR systems is in customer service and helpdesk and varies across various industries like banks, automobiles, telecom, consumer electronics and many more.

2. Problem Statement

The IVR systems have evolved over time. Integrated with enterprise apps, customer support agents can address customer issues and concerns with greater confidence. The integrated IVR systems makes customer context available to the customer service agent which helps address customer issues and concerns in a better way. However this does not scale always resulting in longer wait time for customers, call switching and call transfers adding to customer frustration and insufficient or inadequate domain knowledge to address concerns or queries.

The IVR systems are also rigid in terms of services that are provided to end customers. Hierarchical menu selection to get to the desired service level is always a pain point for customer.

In this white paper we will discuss about the transformation for customer interaction using voice agents without altering the traditional ways i.e. users call in for help. This works great with old as well as younger user-base as it makes use of existing infrastructure like telephone systems and users need not adopt and learn about any new technology to reach out for support.

3. Proposed Solution(s)

a. Introduction of Solution and Advantages

Cognitive voice agents are the future of efficient user interaction, a natural evolution of IVR systems and the preferred user interface of 21st century. Voice agents will

- Always be available
- Scalable to handle increase in load in a transparent manner resulting in zero or minimal wait time for calling in customers
- Capable of carrying out conversation on their own without requiring help from human agents
- Learn from past interactions and get better over-time
- Transfer call to human agents when customer demands
- Assist human agents with real time transcriptions for better responses in real-time

Building intelligent voice agent requires

- Always be available
- Ability to extract intents and entities from natural language
- Ability to execute action based on intent and entity to generate a response. The action could be embedding interactions with internal/external APIs, services to add specific business and domain knowledge or information
- Convey response back to user in natural language
- Training and customizing voice agents based on your domain

We will discuss three approaches on building / using voice agents using IBM Watson cognitive services. The advantages and limitations for each of the solution are detailed in subsequent section

- **Fully cloud hosted solution:** Voice agents using Twilio service on IBM Bluemix
- **Leverage your existing IP-PBX infrastructure:** Voice agents using IP-PBX solution and SIP stack
- IBM Watson Voice Gateway

Above approaches revolve around making use of Watson's cognitive APIs to consume and emit audio steams in real-time over SIP.

The voice agent uses below IBM Watson Cognitive services

- **Speech to text (STT):** Convert incoming audio streams over SIP into textual representation (transcription)
- **Conversation:** Extract intents and entities from transcript. Determines response based on intent, entity and context of the user conversation. The conversation service is trained in specific domains or skills to converse with the user. Note that training the conversation service and training corpus creation is out of scope of this white paper.
- **Text to speech (TTS):** Convert response into audio stream to be sent back to user over SIP and eventually relay over SIP phone or PSTN line

b. Solution Details:

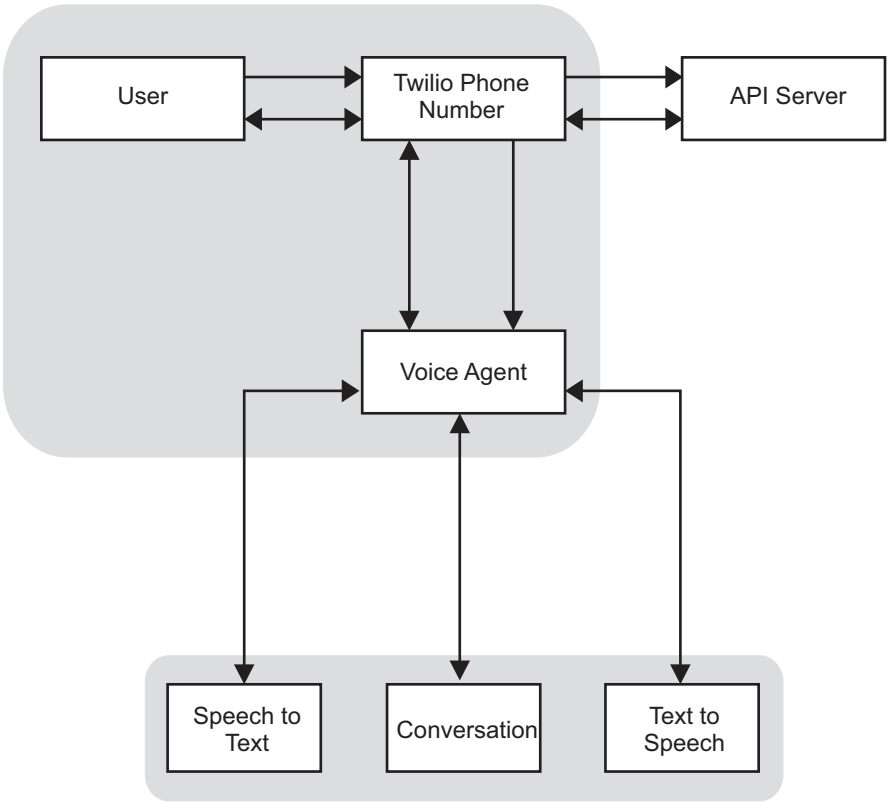
1. Voice agents using Twilio service on IBM Bluemix

This solution is fully cloud based and leverage Twilio service on Bluemix. Twilio service provides global PSTN numbers and SIP endpoints that are used to build voice agents. The system is composite and resides on cloud; no infrastructure setup activities required.

The solution stack consists of

- Twilio service on Bluemix
- Custom API server and SIP endpoint on Bluemix or in any cloud
- Watson cognitive services for speech-to-text, conversation and text-to-speech

A high level overview is shown below



Components

User:

A phone on any network

Twilio platform:.

Cloud communications platform for building SMS, Voice & Messaging applications on an API built for global scale. The solution use global PSTN number and SIP endpoint provided by Twilio

API server:

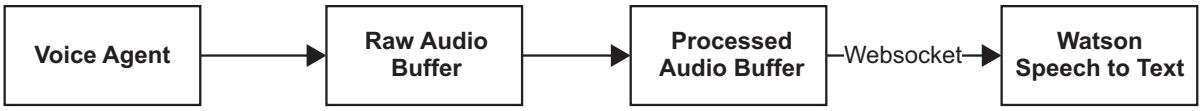
Handles incoming PSTN call and routes to voice agent for processing. The API server is hosted on a computer with publicly visible IP address.

Voice Agent:

Receives audio stream over SIP. Uses Watson speech-to-text to convert audio stream into transcript. The transcript is passed to Watson conversation to extract intents and entities. Based on the intent and entity, a response is created. The voice agent is then responsible for sending voice data to Twilio which forwards it to user's phone.

The voice agent is hosted on a computer with publicly visible IP address. It can be collocated with API server. The component is configured with settings and credentials provided by SIP provider, in this case Twilio. However any SIP provider can be used including self-hosted SIP solutions.

Speech-to-text flow:



Workflow:

1. User dials the PSTN number provided by Twilio.
2. Twilio looks for configured webhook (API server) for that number and makes a request to webhook.
3. API server returns a TwiML response telling Twilio to dial the SIP endpoint
4. Twilio dials the SIP endpoint and sets up a conference between user and the voice agent
5. Voice agent then gets ability to carry full duplex audio communication with user's phone
6. Audio data from user's phone is passed through following API chain: Watson speech-to-text -> Watson Conversation -> Watson text-to-speech
7. Audio from Watson text-to-speech is sent back to user's phone.
8. Call is disconnected when either party hangs up.

Advantages:

1. Full cloud hosted solution; no infrastructure setup cost and maintenance costs involved
2. Flexible voice agent can manage interaction with external APIs and apps to enrich the response before sending

Limitations:

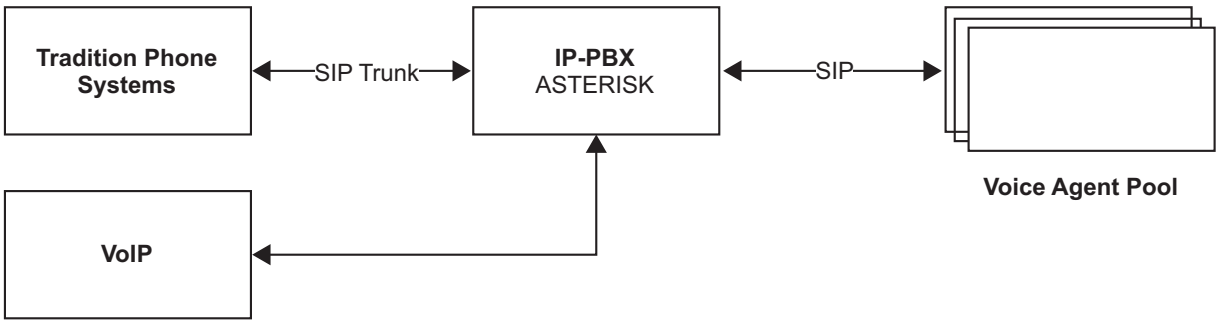
1. The approach uses call recording, recording retrieval services from Twilio. Owing to sequential nature of these services and latency involved in each operation, this approach is not very practical as the conversations will have noticeable delays

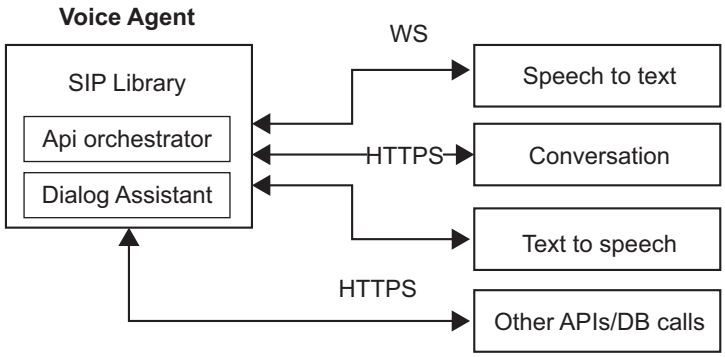
2. Voice agent using IP-PBX solution and SIP stack

Most of the companies have IP-PBX system installed. This option is to leverage the existing IP-PBX solution to deploy voice agents in your infrastructure. The solution stack is built using

- Asterisk as IP-PBX system
- Twilio for SIP trunk.
- Watson cognitive services for speech-to-text, conversation and text-to- speech

A high level overview is shown below





About Twilio SIP trunk:

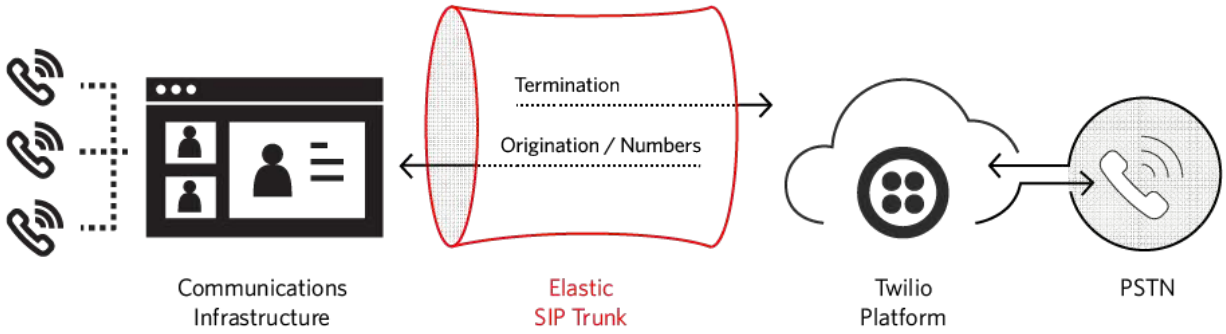
Before looking at various components, it is important to understand Twilio SIP trunk capabilities and how it works.

Twilio SIP trunk allows connecting a PSTN number to organization's IP telephony infrastructure defined by origination and termination settings.

Origination setting dictates how to handle incoming calls to the PSTN number. Twilio SIP trunk forwards the call to sip:{pstn_ number}@{add ress} which must be configured on Twilio dashboard. Here, {address} points to organization's PBX IP or hostname and {pstn_number} is the number provided by Twilio. When a person makes a call to PSTN number, the call is digitized and passed to organization's PBX as a SIP call which then gets sent to Watson voice agents.

Termination setting dictates outgoing calls. A trunk is configured in PBX that points to our termination URI such as {example}.pstn.twilio.com.

Twilio elastic SIP trunk offers unlimited concurrent calls by scaling automatically.



Source: https://www.twilio.com/docs/api/sip-trunking

Components :

IP-PBX:

IP PBX is a private branch exchange that manages internal telephony of an organization and works on top of Internet protocol; helps reduce communication costs drastically.

In the above case, the IP-PBX acts as a gateway between voice agents and external networks; both IP based or traditional PSTN telephone networks.

This solution uses Asterisk IP-PBX solution. Details of Asterisk stack installation and configuration can be found in Appendix – B. The solution can be enhanced more using following features of Asterisk

- SIP trunk
- WebRTC support for SIP calls
- Call transfer and queueing
- Custom dial plans and extension rules to allow public or private telephone systems to connect with a voice agents

Voice agent pool:

A pool of voice agents waiting to engage with users.

Voice agent:

Voice agent is a SIP endpoint built on top of existing SIP libraries such as JSIP or peers-lib and comprises of an API manager which orchestrates API calls between multiple services offered by IBM Watson and a dialog assistant that helps streamline conversation dialogs to organization's needs.

Orchestrator:

The Orchestrator choreographs multiple APIs of Watson such as speech to text, text to speech

Dialog Assistant:

Interface with Watson conversation API and incorporate responses with business specific intelligence and data before passing the response back to API manager.

For example; Watson conversation API may be configured to return the following response:

```
Your next dental checkup is scheduled on {next_schedule_date}
```

Here Dialog assistant can fetch the schedule date from a database and embed the date in dialogue.

Workflow:

1. Customer places a call to a PSTN number for a service request or assistance.
2. The PSTN number is configured with a SIP trunk which forwards the audio streams from tradition phone system. This is dictated by the origination scheme of your SIP trunk setup.
3. SIP trunk connects with an organization-wide IP-PBX which then forwards SIP traffic to endpoints (Voice agents).
4. A Voice agent (SIP endpoint) forwards audio stream to Watson speech-to-text service over a web-socket connection and receives text transcriptions in real-time.
5. When a pause in speech is detected, the transcriptions are sent to Watson conversation service to fetch the next dialogue.
6. The dialogue is parsed and processed further according to business requirements; dialog assistant has access to dialogue transcriptions, intent, entities, confidence score and alternative transcriptions. Third party APIs, database calls can also be incorporated at this stage to further enrich dialogues.
9. Once the final dialogue is fabricated, the text is send to Watson text-to-speech instance and audio stream is received and forwarded to SIP endpoint.

Advantages:

1. Built on top of existing SIP stack and libraries
2. Closely coupled with own IP-PBX such as Asterisk or 3CX, hosted on premise or in the cloud. Most organizations have some form of organization-wide PBX already setup.
3. Scale up or scale down infrastructure and number of voice-agents at any time
4. Multi-tenancy by connecting one or more direct inward dialing, as defined in asterisk configuration files, to Voice agents or by having a pool of Voice agents waiting to engage with customers
5. Flexible plug-in architecture: Ability to replace Watson Speech to text, Watson Text to speech with other services as per business requirements. Voice agents are designed to be very modular. One can extend and override voice agents to utilize other services for Speech to text or Text to speech. For example: by extending the method void speak (String text); one can incorporate another TTS service or SDK such as FreeTTS or espeak.

Scalability of the solution:

In order to scale the solution, the components themselves has to be scalable. The scalability aspects of few components are discussed below

1. **Twilio SIP trunk:** Twilio's Elastic SIP trunk scales automatically and supports unlimited concurrent calls. Twilio SIP trunk supports multiple origination URI for efficient load balancing. Calls are forwarded to the origination URI based on priority and weight of the URI. This allows association of multiple PBX instances and therefore multiple voice agents - https://www.twilio.com/blog/2015/07/sip-trunking-load-balancing-and- failover-made-easy.html
2. **Asterisk IP-PBX**: Asterisk PBX can be deployed on a scalable infrastructure such as EC2 or IBM Softlayer which supports seamless scaling. Asterisk PBX hosted on premise can be scaled by providing more hardware and software resources
3. **Watson cognitive services** (STT, TTS and Conversation): The Watson cognitive services are cloud hosted services on IBM Bluemix and are scalable.
4. **Orchestrator:** No heavy lifting is done by orchestrator; it delegates calls to respective component. Orchestrator can be vertically scaled easily. The horizontal scaling is possible via router/load balancer and as such orchestrator does not maintain any state which might impact horizontal scalability.
5. **Dialog Assistant**: The dialog assistant is domain specific and performs the function of enriching the response by connecting to other internal/external APIs and/or services. The dialog assistant has to be carefully designed so as to not introduce great latency during enrichment of the response.

Multi-tenancy of the solution:

Multi-tenancy can be achieved by using these two features:

a. Multiple origination URI configured in Twilio dashboard

b.. Asterisk's ACD - Automatic call distribution queues

Twilio Multiple origination URL:

Twilio places forwards calls to origination URI with lowest priority. If SIP session isn't established, due to busy endpoint or any other failure, then Twilio dials the next origination URL

Asterisk's AC D - Automatic call distribution queues:

Asterisk's ACD - Automatic call distribution queues where a pool of voice agents wait to answer calls and incoming calls are queued, usually in a FIFO order, and assigned to next available voice agent.

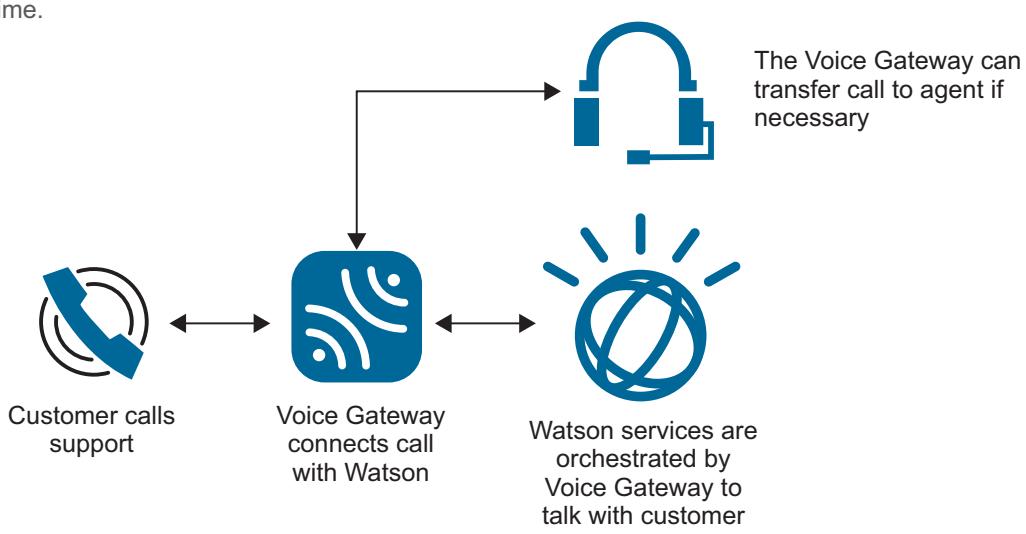
Limitations:

1. Twilio does not support SIPS (SIP secure) and only supports G.711 audio codec which means other efficient audio codecs like G.722 or G.729 cannot be used

3. IBM Watson Voice Gateway

Voice Gateway for Watson is offered as a complete solution to deploy voice agents powered by IBM Watson's APIs, with two different modes:

1. Cognitive self-service agent: capable of full duplex interaction with customer without intervention of human agents
2. Cognitive agent assistant: assist a human agent by transcribing and analyzing conversations in real-time.

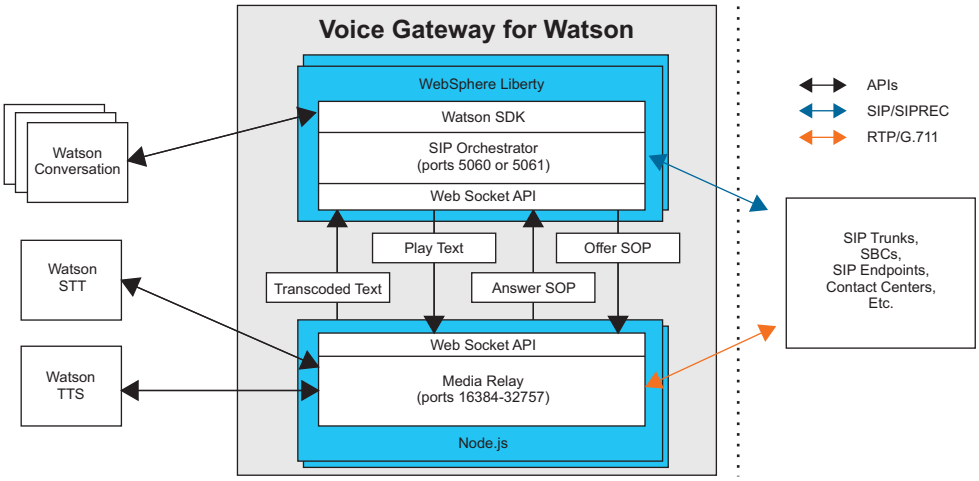


An overview of Voice Gateway for Watson depicts how multiple APIs of Watson are orchestrated together into a complete solution.

Two important layers of the Voice Gateway are:

SIP orchestrator: Communicate with SIP endpoints -> SIP invite, SIP REFER; session border control; API calls to fetch new dialog from Watson conversation service

Media relay: relay audio stream to/from SIP endpoint to/from Watson audio services i.e. STT and TTS. Protocol/codec: RTP/G.711



Advantages:

1. Ready-to-deploy containers with minimum configuration
   a. Docker image
   b. IBM containers for bluemix
2. Robust, well documented, developed at IBM
3. A generic solution offering several features
   a. Transfer call to human agent, allow users to barge-in
   b. Hang-up, music on hold, Control interaction via state variables embedded within Watson conversation responses.
   c. Latency auditing
   d. Audio recording, DTMF support

Limitations:

1. Limited development and production use
2. Multiple tenancy not supported yet
3. Closed source, proprietary software, limited community and support
   https://www.ibm.com/support/knowledgecenter/SS4U29/limitations.html
4. Enriching the response is not supported out of the box, however the feature is under development. A workaround can be used where the WATSON_CONVERSATION_URL parameter of Voice gateway could point to REST proxy which will forward the conversation request to Watson; then receive a response and modify it and pass back to Voice gateway.
   https://developer.ibm.com/answers/questions/356108/customizing-voice-gateway-sip-orchestrator/

4. Challenges with Voice Agents

Although voice agents aid with customer service and alleviate a few pain points, there are certain challenges that needs to be addressed to have a proper solution in place. As the systems get mature, many of the challenges listed below will be overcome or fixed by right strategy. Note that some or all of the challenges below also applies to the solutions listed above

1. No visible UI over telephone:
   a. No visual cues to indicate when to start talking, when to stop taking and when voice-agent is thinking. Example: When talking to Google Now or Siri, users can observe subtle visible cues indicating different states of voice agents like when it is ready to listen, when it is listening, when it is processing the previous utterance etc. A GUI also makes presenting some data formats, like tables, easier.

2. Accuracy of speech-to-text over traditional phone systems
   a. Audio quality: traditional phone systems use G.711 codec and supports single channel, narrowband. The sampling rate is low with of 8KHz sampling and 64 kbps bitrate.

b. When low quality audio is passed to Watson, there are some issues with speech to text for small sentences with little context. Example: Watson STT may confuse between homophones (similar sounding words) during transcription process. However this can be tackled to some extent by training custom speech models specific to one's business domain
https://www.ibm.com/watson/developercloud/doc/speech-to-text/custom.shtml#addCorpora

c. Background noise: Background noise can affect the transcript process. Natural language understanding process can report a lower confidence thus resulting in improper response or no response

d. Background speech: Natural language understanding works best when single person is speaking. Cases when more than one person is talking simultaneously, can result in lower confidence during transcript process. This can affect further processing in terms of identifying intent and carrying out proper action. This will result in either improper or inaccurate response or no response.

3. Quality of service:

a. Keep dialogue latency to minimum for natural sounding conversation. Latency plays a big role in determining QoS of a solution, such as this, which is based on natural sounding turn-based conversations between a human (the customer) and a computer agent.

We referenced study on “pauses and gaps during conversation”
http://www.speech.kth.se/prod/publications/files/3418.pdf

During our test calls an average delay of 500-800 ms between dialogues was found to be acceptable by English speaking users. Several studies suggest that a delay below 1 second is optimal for natural sounding turn-based conversation and too many gaps between dialogues confuse users and add to frustration.

b. Subtly dictate the flow of conversation. Voice agents aren't good at handling off-topic conversations whereas humans often like to talk about diverse topics including making small talk. This scenario must be handled carefully without sounding too robotic/programmed. This can be achieved by using intent and confidence score emitted by Watson conversation service to detect off-topic dialogues and then gently guiding user back to last relevant dialogue. The conversation service can also be trained to make small talk.

c. Understand the tone of conversation (emotions such as frustration or anger) and act accordingly. Understanding emotions: traditional IVR and voice based systems do not understand human emotions at all. People, on the other hand, can display a variety emotions during a conversation such as anger , frustration, joy etc.

Watson Tone Analyzer service can be used to detect the emotion in such a scenario and build a response accordingly. For example, if a customer with repeated angry dialogues can be transferred to a human agent. https://www.ibm.com/watson/developercloud/doc/tone-analyzer/index.html

All of these parameters can make or ruin a good conversation experience and are essential in determining QoS.

4. Future Direction / Long-Term Focus

The voice agent will evolve with time with advancements in machine learning related to speech- to-text, conversation and text-to-speech. Learning from the past conversations and domain, process specific training will be important aspects of any voice agent. A good strategy is to start with one domain and expand to other domains as voice agent matures over time. The domain knowledge, process knowledge per say is not constant, which requires voice agents to be trained periodically for effective interaction.

The voice agent needs to orchestrate between Watson cognitive services and others APIs and services to build a response. Thus response latency is a key consideration. Any higher latency in response may result in end user thinking unresponsive behavior can lead to frustration. Appropriate measures have to be put in place to reduce the inconvenient gaps during conversations.

Voice agents have ability to transfer call to human agent, however vice versa is not orchestrated. Too many transfers can again lead to the same situation of long waiting time, inappropriate or inadequate response because of lack of domain knowledge or process knowledge on part of human agent. It is increasing important to plan right mix of voice agents and human agents and build capacity accordingly.

5. Results / Conclusion

The voice agents are here to stay and their usage will rise over time. The recommended solution depends on your existing infrastructure or your plans and investments in such infrastructures. Persistent Systems Ltd can help you do an assessment and help you build right voice agent for your business case. Persistent Systems Ltd has extensive experience and case studies building various types of bots for various domains and industries including and not limited to automobiles, healthcare, HR and finance.

Appendices

Appendix A – Scenarios

The following scenarios have been referenced in this document for building and deploying voice agents

- Fully cloud hosted solution: Voice agents using Twilio service on IBM Bluemix
- Leverage your existing IP-PBX infrastructure: Voice agents using IP-PBX solution and SIP stack
- IBM Watson Voice Gateway.

Appendix B – Installation, configuration and deployment

Please contact authors in Appendix C to get access to assistance, detailed instructions and/or source code

Appendix C – Authors

Jaydeep Ayachit
jaydeep_ayachit@persistent.com
Sr Architect, Persistent Systems Ltd

Manish Raj
manish_raj@persistent.com
Software Engineer, Persistent Systems Ltd

Appendix D – References

https://www.linode.com/docs/applications/voip/install-asterisk-on-centos-7
https://www.ibm.com/watson/developercloud/tone-analyzer.html
https://wiki.asterisk.org/wiki/display/AST/Asterisk+ Architecture
https://www.twilio.com/docs/documents/14/AsteriskTwilioSIPTrunkingv2_0.pdf
https://cjcarpen.gitbooks.io/voice-gateway-for-watson/
https://cjcarpen.gitbooks.io/voice- gateway-for-watson/deploydocker.html
https://www.ibm.com/support/knowledgecenter/SS4U29/welcome_voicegateway.html
https://www.ibm.com/us-en/marketplace/voice-gateway
http://www.asteriskdocs.org/en/3rd_Edition/asterisk-book-html-chunk/asterisk-ACD.html

PERSISTENT

About Persistent Systems

Persistent Systems (BSE & NSE: PERSISTENT) builds software that drives our customers' business; enterprises and software product companies with software at the core of their digital transformation. For more information, please visit: www.persistent.com

India
Persistent Systems Limited
Bhageerath, 402,
Senapati Bapat Road
Pune 411016.
Tel: +91 (20) 6703 0000
Fax: +91 (20) 6703 0009

USA
Persistent Systems, Inc.
2055 Laurelwood Road, Suite 210
Santa Clara, CA 95054
Tel: +1 (408) 216 7010
Fax: +1 (408) 451 9177
Email: info@persistent.com