Voice Search
Benchmarking Report

SLANG LABS

Comparison of Slang Retail Assistant with Google Voice Search

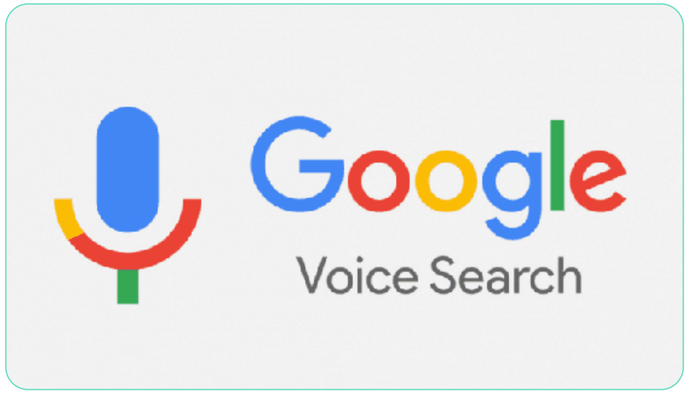
Table of Contents

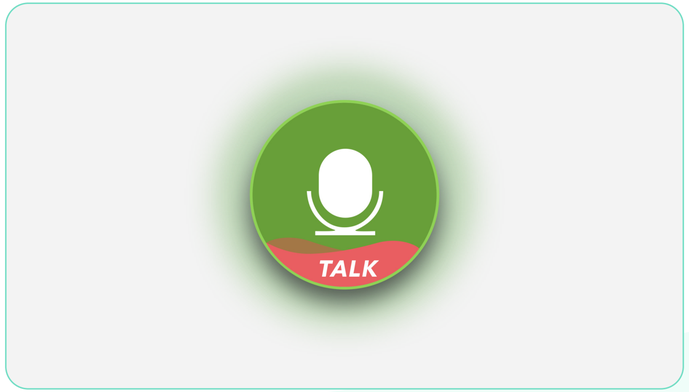
Abstract

More and more brands are looking to add voice-search functionality to their mobile and web apps to make them accessible by a larger number of users, as well as to increase the productivity and efficiency of existing users. However, it is not very clear how to evaluate the various voice-search offerings available, their trade-offs and which offering would provide the best value to these brands. This benchmarking report attempts to provide some parameters based on which voice-search offerings could be evaluated, and also provides a comparison of two popular voice-search offerings for retail brands, based on these parameters.

Background

The Slang Retail Assistant is a domain-specific, pre-built Voice Assistant that easily allows retail brands to add voice experiences, including voice-search to their mobile and web apps. The Slang Retail Assistant is optimized for the retail domain and has the ability to recognize product types, variants, categories, SKUs and other details specific to the retail domain accurately. It also uses a sophisticated NLU engine to precisely map voice utterances for search into search queries. The Slang Retail Assistant is currently being used by multiple retail brands in India.





Background

While there is no distinct offering from Google called Google Voice Search, many Android apps commonly implement a voice-search widget using the SpeechRecognizer API available as part of the Android platform. The SpeechRecognizer API is a speech-to-text API that collects voice utterances from users and converts them into text using Google's speech recognition service. It is easy to use and available free of charge on Android devices, so brands have already integrated it into their apps, although it is not specialized for search or the retail domain. This widget and the recognition service powered by Google will be collectively referred to as Google Voice Search in this report.

Since both of these offerings provide voice-search functionality, are easy-to-integrate and popular, brands are often confronted with the question of which offering to integrate with their app. One of the most common questions that we've heard from our customers at Slang Labs is how Slang Retail Assistant compares, relative to Google Voice Search. There have been no comparative studies between these two offerings and the impact they could have on voice-searches performed on the brands' apps. So in this report, we will perform a comparative analysis between these two offerings and try to answer these questions.

Experiment Setup

For the comparative analysis, we designed an experiment where we could supply the same input to both voice-search offerings under identical conditions, collect the output, measure the performance of each offering and compare the performances. There were three parts to experiment:

A. Data collection

B. Running the experiment

C. Analyzing the results

A. Data collection

In order for the experiment to produce reliable results, we needed to gather inputs that would be representative of inputs that would be supplied to the voice-search offerings in the real world. To achieve this, we gathered a large number of voice samples from individuals across India, with the following characteristics:

A. English utterances of 100 popular retail items as they would be spoken into a voice-search engine.

B. Speakers distributed across 10+ states of India to cover variability in dialect and accent.

C. Utterances distributed across varying lengths, descriptiveness and verbosity.

We gathered around 3000 samples of audio data that matched the above characteristics, from 40 unique speakers, and used them for
the experiment.

B.Running the experiment

We needed to run the experiment in a way that resembled real-world usage as closely as possible, while also ensuring minimal differences in the environment between each leg of the experiment. To achieve this, we chose to use an open-source Android app with both voice-search offerings integrated, along with a driver script to automatically provide inputs and collect performance results. The details are below:

Host: Mac Mini with 16 GB RAM, running Mac OS Big Sur, connected to an external speaker via Bluetooth

Device: Samsung SM-A260G with 16GB RAM, running Android 8.1.0

App: Open-source grocery store demo app (VAMO) with both Slang Retail Assistant and Google Voice Search integrated running on the device. The search database on the app was populated with all the 100 items that were used for the audio data collection, so that a correct search query would always return at least one search result.

Driver: A python script responsible for invoking voice-search in the app, providing voice input and collecting results from the app

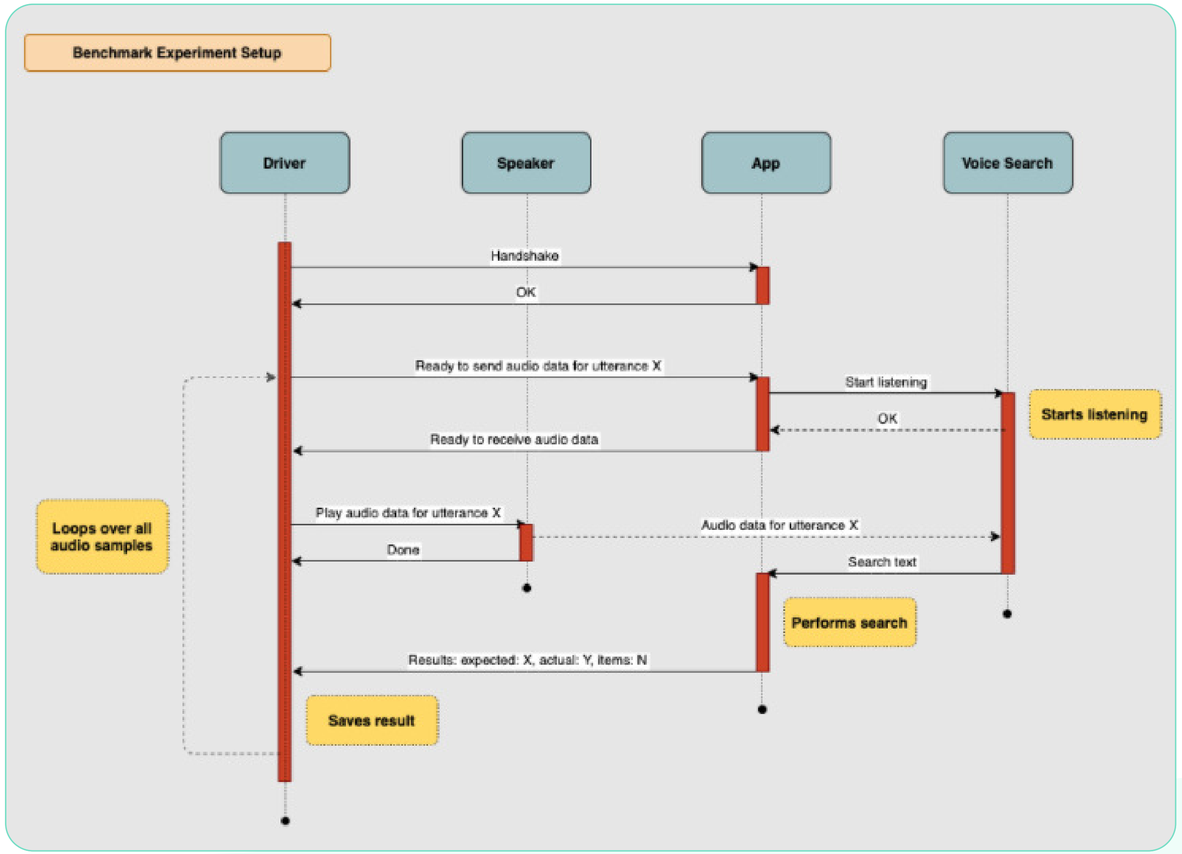
B.Running the experiment

Integration: For running this experiment, we built a driver script that would run on a host and communicate with the app running on the device also connected to the same host. The driver communicates with the app in the following way:

1. The driver initiates a handshake with the app to start the experiment
2. The driver sends a message to the app indicating that it is ready to start sending audio samples to be processed by the app via voice search
3. The app initiates voice search, which starts listening for audio input and sends back a response to the driver
4. The driver then plays the appropriate audio clip on the connected speaker, which is picked up by the voice search
5. The voice search is processed: the speech in the input sample is converted to a search query and passed to the app's search field and the actual search is performed by the app
6. The app collects the results from the search and sends back the results to the driver for further processing.

B.Running the experiment

The sequence diagram shown below helps display the interactions between the various components visually.



C. Analyzing the results

Once the experiment was run and the results for both voice-search offerings were collected, we had to formulate a scoring formula, assign scores to each result using this formula and compare the final scores.

C.1 Structure of the results

The results sent back from the app to the driver consists of the following information

A. **ID:** the id of the input sample that was processed
B. **expected:** the expected transcribed value of the utterance
C. **actual:** the actual transcribed value of the utterance
D. **search_term:** the search term that was passed to the app's search
E. **item_count:** the number of items present in the search results

C.2 Scoring Formula

The scoring formula needs to accurately reflect how well the voice search performs in converting an input utterance into a valid search query for the app, which further results in a successful search. To achieve this, the formula needs to factor in two abilities of the search engine:

1. Accuracy of speech recognition:

Accurate speech recognition is the first key step in correctly converting an input utterance into a useful search result. Therefore, the scoring formula needs to give importance to this ability of the voice search offering. For example, it may be common for a speaker to pronounce the term “corn flakes” in a way that sounds like “cornflex”, but it’s important that the voice search recognizes it as “corn flakes” for the subsequent search to be useful.

2. Effect on search quality:

Once the search term is transcribed accurately, the next important requirement is for the search term to be passed down to the app in a way that is optimal for searching. For example, it’s natural for a user to say “show me organic onions” while trying to search via voice. But passing the entire utterance down to the app would likely result in suboptimal search performance because search engines are unlikely to be optimized for extraneous words such as “show me”

C. Analyzing the results

With these requirements, the following scoring terms and formula have been proposed

1. Speech Recognition Score (SRS): This score is a value between 0 and 1 that captures the accuracy of speech recognition. It is calculated by computing the Levenshtein distance between the expected utterance and the actual transcription, normalizing it by the length of the expected utterance and amplifying the resulting value.

2. Search Quality Score (SQS): This score is a value between 0 and 1 that captures the effect of voice search on overall search quality. It is based on the assumption that each possible input utterance should result in at least one search result being returned by the app upon successful search. It is calculated to be equal 1 if the voice search produces at least 1 search result and 0 if no search results are produced.

3. Voice Search Score (VSS): This is the final score that is a function of both SRS and SQS. The idea behind combining these two scores is that a search is likely successful only when the voice search performs well in both categories and not just one. For example, “besan flour” could be transcribed as “basin floor” and could result in one or more results that match with “basin” or “floor”, but this is not the optimal result expected from the search and combining the two scores will lead to a reduction of such false positives

C. Analyzing the results

The overall scoring formula can be visualized below:

$$CER = distance(expected, actual)$$

$$NCER = \frac{CER}{|\ expected\ |}$$

$$SRS = (1\ - NCER)^2$$

$$SQS = \begin{cases} 1 & if\ |\ results\ |\ > 0 \\ 0 & otherwise \end{cases}$$

$$VSS = SRS \times SQS$$

Results

Applying the scoring formula described in Section 3.2 on the results obtained by both voice-search offerings gives us the following results (Raw result data can be found here: <TBD>

Google Voice Search	Slang Retail Assistant
Avg SRS: 0.77	Avg SRS: 0.86
Avg SQS: 0.61	Avg SQS: 0.89
Avg VSS: 0.54	Avg VSS: 0.79

Based on the Avg VSS scores obtained above, we can conclude that the overall voice-search performance of Slang Retail Assistant is around 46% higher than that of Google Voice Search.

Summary

In this report, we have described the concept of Voice Search in mobile/web apps and introduced two of the key voice-search offerings in the industry today: Google Voice Search and Slang Retail Assistant. We then motivated the need for an objective comparison of the performance of these voice-search offerings, followed by the description of the data collection, experiment setup and scoring formula. Finally, we looked at the scores obtained by running the experiment and attempted to reason about the causes for observed performance characteristics.

In the future, we plan to experiment with some enhancements to the scoring formula, such as deriving baseline SQS using an industry-standard search engine. We also plan to release the source code for the entire experiment setup, including the app, the driver and scoring scripts. We also look forward to running similar voice-search comparisons between other popular voice-search implementations.