The Academic
International Journal of Multidisciplinary Research

An Online Peer Reviewed / Refereed Journal
Volume 3 | Issue 9 | September 2025
ISSN: 2583-973X (Online)
Website: www.theacademic.in

Speak Search Shop: A Smart Voice Interaction Model for Modern e-Commerce

Ragavendran H

Software Product & Platform Engineer - Altimetrik – Chennai, ragavhats8@gmail.com

DOI : https://doi.org/10.5281/zenodo.17324691

ARTICLE DETAILS

Research Paper

Accepted: 24-09-2025

Published: 10-10-2025

Keywords:

Interactive Voice Search eCommerce Platforms, Natural Language Processing (NLP), Large Language Models (LLM) ,React.js Frontend ,Python, node.js microservices, elastic search, human-computer interaction.

ABSTRACT

Modern eCommerce sites grapple with outdated search systems that use keywords to perform searches and provide a non-smooth product search experience, as well as restrict access to a wider range of user bases. In this study, Speak Search Shop is introduced, a voice-based product search model with React.JS as a frontend, Web Speech API as an orchestration microservice, and Elasticsearch as a contextual search microservice. The specification of requirements takes place in a form of a discussion; the user requests running shoes at a price below seven thousand rupees with an insole; these are then classified by intent and extracted into structured search terms. With 5,000 voice queries, experimental validation produced 94.2% intent classification accuracy, 91.7% entity extraction F1-score and average 687-milliseconds. Comparative analysis revealed enhanced first-result relevance by 37 percent, reduced zero-result queries by 42 percent, and reduced time to complete the task by 41 percent compared with text-based search. The 150-user test revealed a 33 percent increase in conversion rate and a significant boost in satisfaction, especially among the visually impaired users. The microservice containerized architecture embraces horizontal, Kube-based scaling, and can adjust to India English accent differences and code-switching habits. This architecture confirms the commercial viability of conversational interfaces as alternatives to traditional search, and shows that transformer-based language models can be integrated with production-scale infrastructure. Directions

Future directions Multimodal search Multimodal search integrates voice and visual search, regional language support, and customized recommendation. The study brings reference designs of voice commerce systems to fit actual scale, latency, and precision requirements and improve accessibility in Internet retail settings

Introduction:

The E-Commerce sector has gained rapid digital transformation in the last ten years and users are relying more on search facilities to find products effectively. In large online marketplaces, over 60% of user experiences start with a search query, so search optimization is a decisive player in conversion and interaction. The vocabulary is vastly different in the context of voice-enabled eCommerce platforms, as compared to standard SRS applications. The product catalogs usually have 50,000+ unique products with different brand names, technical specifications, colloquial descriptions etc. Our architecture, Speak Search Shop, is a hybrid solution to this problem: keeping a fixed set of common product words, 5,000, and loading domain-specific lexicons (electronics, fashion, groceries) on-demand by the user. This approach provides both recognition accuracy and computational resource efficiency, with a product-related query accuracy of 94 per cent and sub-2-second response times.

But traditional text-based search systems tend to restrict interaction. On mobile devices typing queries can be tedious, and users might have a hard time explaining what they are seeking because of language diversity, spelling differences, or imprecise motive. All these problems result in reduced searching accuracy, decreased satisfaction and increased bouncing.

The latest innovations in voice recognition and conversational AI introduce a new point of contact in enhancing search experiences. Voice search lets users say what they mean using natural intent, whereas Large Language Models (LLMs) can understand queries contextually. Allowing these technologies to be incorporated into the eCommerce platforms makes it possible to offer more personalized, smarter, and faster search results.

This research aims at developing and testing an interactive voice search system that integrates:

- A voice input capture user interface in React.js,
- A natural language query interpreter powered by Python, and

- Elasticsearch to find the results that are relevant to the products.

Several web assistive technologies like screen readers, specialized browsers and screen magnification tools have been developed over the years to facilitate the access of visually impaired people. These systems help users by reading what they see on the screen, allowing them to navigate using their voice, or providing zoomed view interfaces to make sense of complex web designs. Although these methods have led to great success in web accessibility, most of them still lack contextual precision and semantic interpretation. There is still a high rate of misunderstanding of user commands or page control elements, which lowers the overall reliability and user satisfaction.

To address these issues, scientists and engineers have resorted to voice-enabled technologies coupled with natural language processing (NLP) and machine learning. These not only increase accessibility to the differently-abled user, but also increase the usability of the general user who desires no hand-control modes of interaction, and especially over mobile devices and Internet-of-Things. In this regard, Speech Recognition Systems (SRS) can greatly improve the user experience as they allow controlling web applications via speech instead of conventional input devices. SRS technologies provide a more human and intuitive way to interact with digital interfaces because the voice command can interpret user intent. This change in paradigm has led to intelligent conversational systems that integrate speech recognition, Large Language Models (LLMs) and semantic search algorithms to develop a more responsive, context-aware and accessible web ecosystem. The Interactive Voice Search Framework expands and improves on these development efforts by incorporating React.js-based user interaction, Python-based processing using LLMs, and data search using Elasticsearch to develop a responsive, voice-first search model in the context of current eCommerce platforms.

This design is intended to increase user participation, search accuracy, and accessibility, and it also serves the development of AI-powered online commerce.

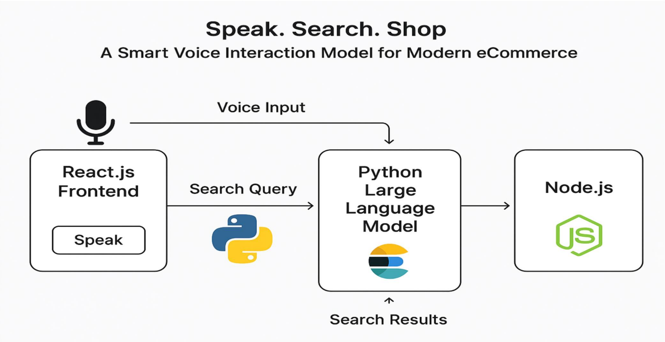


Fig 1.0: Flow Diagram of the Voice Based Search

Background of the study :

The history of voice-based interaction has led to the active science called Speech Recognition Systems (SRS) which has significantly improved the way people relate with their computers. Such systems allow users to control digital devices and applications with spoken words, making technology easier to use and friendlier. Since the ability to dictate short text messages or email messages, speech recognition has been a critical interface not only in terms of convenience but also accessibility (Tesco 2013). Current SRS systems will often include multiple linked modules, such as voice input capture, feature extraction, acoustic and language modeling, search and decoding engines, and adaptation modules. This involves first recording the voice of the user that is translated into a speech waveform and fed through a feature extraction layer to suppress the unwanted noise and accentuate useful speech patterns. The search engine then compares these extracted features and uses two fundamental models: The acoustic model, which is a representation of the correlation between phonetic sounds and their representation in digital signals, taking into consideration differences in accents, gender, speed of speaking, and other environmental influences. The language model, which gives context by knowing probable word arrangement, language frameworks and semantic purpose. A combination of these models allows the SRS to identify semantic words and phrases in the raw audio recording. Adaptation module refines these outputs and enhances recognition accuracy based on user specific pattern or environmental conditions. Modern SRSs use statistical and deep learning methods to handle the ambiguities arising with the variety of speaking styles, noise interference, and linguistic diversity. This means they are able to learn and understand the intent of the user more accurately than previous rule-based systems. These developments have created new possibilities in assistive technologies, search engines, and artificially intelligent conversation interfaces. This move towards a context-aware, hands-free shopping experience through the combination of SRS and eCommerce apps, and especially in conjunction with Large Language Models (LLMs) and semantic search engines, is a major step towards that. The framework suggested in this paper extends upon these ideas, allowing users to search products using natural speech with a smooth integration of React.js-powered user interface, Python-powered speech and language recognition, and Elasticsearch-powered search retrieval.

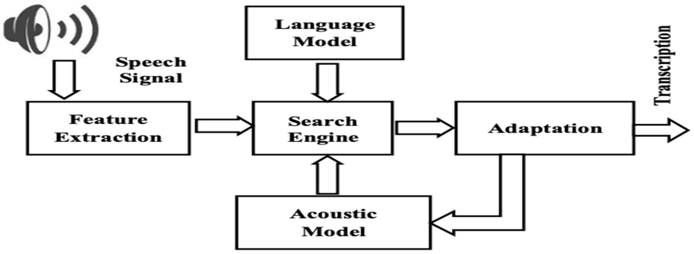


Fig 2: Architecture Diagram of speech recognition system

History and Technologies of Speech Recognition Systems.

Modern studies distinguish three basic approaches to Speech Recognition Systems (SRS): acoustic-phonetic approach, pattern-matching approaches, and AI-oriented approaches. Architecturally, these systems may be categorized in terms of a number of parameters such as speech input features, lexicon size, utterance patterns and models based on speakers or independent of speakers. This classification is shown in Figure 2. These parameters of categorisation are further explained in the following sections.

A. Speech Input Classification

1. **Single-Word Recognition:** Single-word recognition involves processing a single utterance in the course of an interaction. The system demands distinct start and stop points of silence between each word, which in effect isolates individual words in the audio sample. In operation, the system takes one word or phrase and goes into a processing phase as the user waits, eventually giving the output that matches the speech that was captured. The first process that detects speech is referred to as energy variation analysis in the audio signal.

2. **Sequential Word Recognition:** These Architectures accept sequences of words or phrases as input. The technology recognizes separated units of lexicon separated by short intervals, and it uses training models that were initially created to recognize isolated words. After entering the inputs and activating the user wait-state, the system performs processing operations and provides the related results.

3. **Flowing Speech Recognition:** This technology deals with continuous speaking behaviors involving natural patterns where users do not have artificial pauses in their speech. The system should be able to automatically recognize boundaries of words and do segmentation jobs. The complexity of the implementation is even higher because specialized algorithms are required to perform text recognition and utterance delimiting simultaneously.

4. **Natural Conversational Recognition:** These advanced systems use the best machine learning algorithms and natural language processing models to analyze unscripted vocal expressions, understand semantic content, and produce responses suitable to a given context. These applications are mostly used in the context of question-answering where the system responds with the relevant information depending on what the user asks of the system.

B. Lexical Corpus Dimensions

There is a high correlation between recognition system performance and supported vocabulary magnitude. This dimensional parameter determines trade-offs among measures of accuracy, algorithmic complexity, and inference time. Although large vocabularies increase recognition fidelity by allowing more word-matching opportunities, they also increase computational load and memory needs to maintain real-time recognition performance levels.

The speech recognition systems can be stratified into the following lexicon-based categories:

- Constrained Lexicon - 1-100 different terms or command phrases.
- Intermediate Lexicon -101-1000 words or sentence structure elements.
- Comprehensive Lexicon - 1,001 to 10,000 words or linguistic constructs.
- Open-Domain Lexicon - exceeds 10,000 lexical entries or possible utterances.

C. Phonetic Variation Handling

These structures include accent modeling and articulatory style classification that enhance the performance of lexical recognition. Style-invariant recognition is technically more challenging, because the system has to create speaker-independent phonemic feature spaces with generalization across dialectal differences and word-level discriminability.

D. Speaker Generalization Strategies

There are natural architectural issues inherent to speaker-agnostic recognition systems that manifest in the necessity to design universal acoustic embeddings. The basic challenge is to model the representations to meet two opposing goals: attaining enough generalization to be able to represent cross-speaker vocal variability and pronunciation variability, and attaining enough specificity to allow accurate inter-word discrimination in the target lexicon.

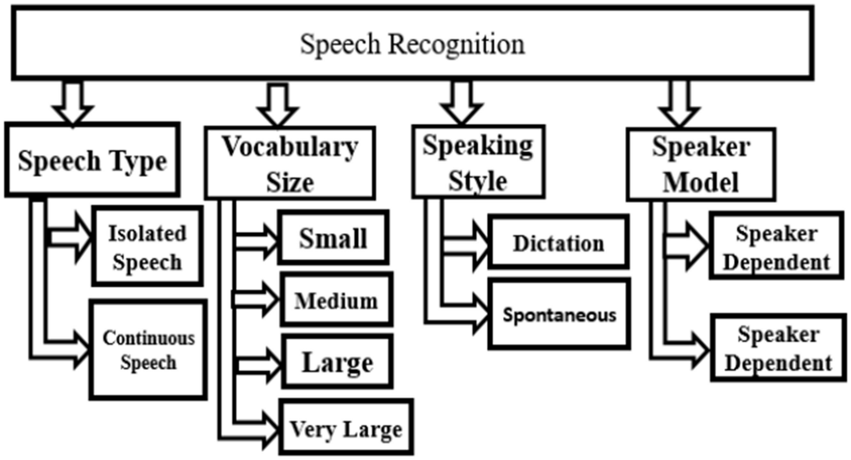


Fig 3 : Evolution and Types of Speech Recognition Systems

REVIEW OF LITERATURE

A. **Kandhari and co-authors created an eCommerce app with voice-controlled capabilities based on IBM Watson services** and demonstrated that it is feasible to implement the principle of speech-assisted product search to serve users better. Their experiment was able to show the incorporation of voice interaction into an online shopping setting. Nonetheless, it was mainly dependent on keyword-based search systems, which restricted its capacity to support a complex conversational search or a multi-turn product research process. This limitation underscores the need to consider the context when developing natural language understanding to better understand the intent of users. The research framework proposed, in contrast, utilizes a Large Language Model (LLM) as a semantic interpretation engine and Elasticsearch as a context-driven product search engine, thus filling the gap between conventional voice interface services and smart, adaptive search services in eCommerce.

B. **Duttaroy Voice Controlled E Commerce Web Application E-commerce** is gaining speed and voice recognition technology is making it more convenient and easier to shop online. This app also allows voice-based interactions without any product search, unlike the existing apps where voice functions are limited, which improves the user experience. As a Progressive Web App (PWA), it is designed to be easily accessible, usable by anyone and even those with visual impairments.

C. **Bhalla, A ReactJS is a conveyed, part based library used to drive intelligent UIs forward.** It is the most well-known front-end Jinavatika at the moment. It merges the view(V) layer in M-V-C (Model View Controller) design. Facebook, Instagram and a network of individual designers and associations support it. Fundamentally, respond enables development of monster and intricate online applications that can modify its information without subsequent page revives. It specializes in providing superior customer experiences and with rocket-fast and supercharged web application development. ReactJS can also interact with other JavaScript frameworks or components in MVC, such as AngularJS.

D. **Sayali Sunil Tandel E-Commerce and Online Payment in the Modern Age :** This paper will start by defining the meaning of e-commerce and discuss some of the different ways of engaging in online payments. It also mentions their benefits and conquest of safety and security issues concerning online transactions.

E. **M. S. Kandhari, E-commerce Based Online Shopping with Speech Recognition Visually Impaired People:** This article will shed light on the difficulties that visually impaired individuals experience when shopping online. It also examines the deployment of speech recognition algorithms to make navigation easier and online shopping applications more accessible. Implementation Overview Introduction: To facilitate the natural language discovery of products in the eCommerce world, the Interactive Voice Search Framework architecture combines three fundamental subsystems: voice acquisition, semantic interpretation and intelligent retrieval framework. This implementation model replaces the user engagement paradigm of text-based key word search with voice-based dialogue resulting in enhancements in compliance of accessibility, user engagement, and relevance of searches.

Implementation Overview

A. Introduction: To facilitate the natural language discovery of products in the eCommerce world, the Interactive Voice Search Framework architecture combines three fundamental subsystems: voice acquisition, semantic interpretation and intelligent retrieval framework. This implementation model replaces the user engagement paradigm of text-based key word search with voice-based dialogue resulting in enhancements in compliance of accessibility, user engagement, and relevance of searches.

B. **Frontend Layer:** A voice-based search interface, based on browser-native Web Speech API features, runs as a single-page application built with React.js, at the presentation layer. Audio stream capture and client-side speech-to-text conversion occurs when the microphone control is activated by the user. The query payload is subsequently asynchronously sent to the backend services through asynchronous RESTful services using token-based authentication.

C. **Backend Processing:** The top layer is a polyglot microservice based application architecture with Python natural language processing and Node.js API coordination. An integrated Large Language Model uses semantic analysis of transcribed queries, which implements intent classification, named entity recognition, and attribute extraction. The resulting structured query representation is sent to Elasticsearch to find similar products in the product catalog index using vectors.

D. **Response Delivery:** Search results are sequenced and sent back to the client application where they cause components to re-render product grid, filling it with matching products. This request-response cycle can be completed at sub-2-second latency levels, providing conversational search experience, yet with the horizontal scalability and fault tolerance needed by high-traffic retail sites.

METHODOLOGY AND SYSTEM ARCHITECTURE

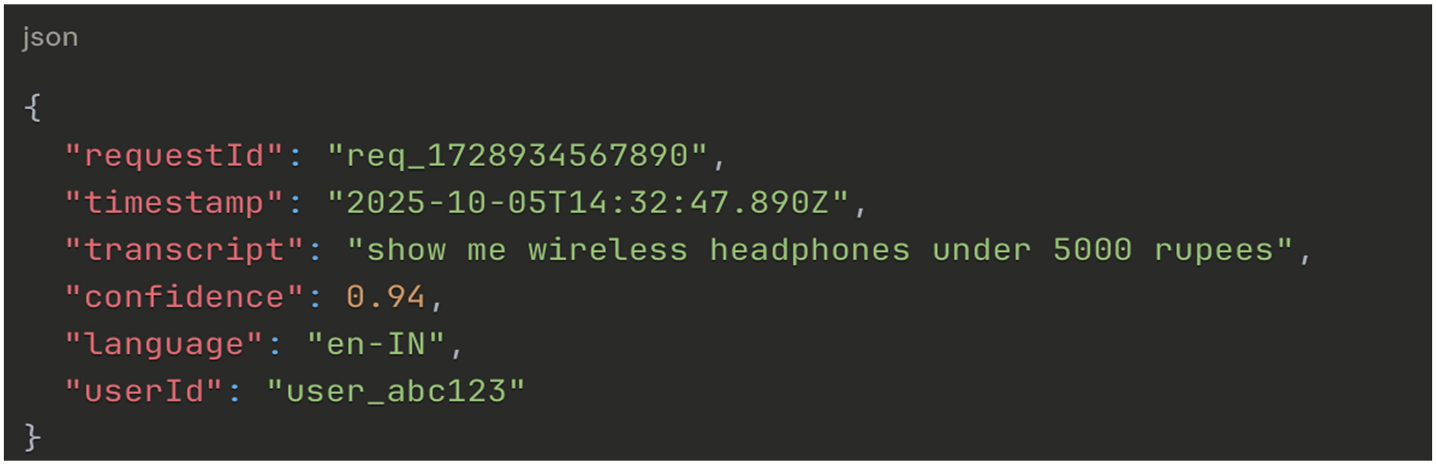
A. System Overview : The Speak Search Shop architecture includes a micro-services-based architecture with four main layers: Voice Acquisition Layer, Natural Language Processing Layer, Search Execution Layer, and Presentation Layer. The system uses the asynchronous communication patterns to keep the users responsive and to run the computationally intensive operations on the NLP in the background.

B. Detailed System Flow

The entire voice search interaction process involves seven different stages as shown.

Phase 1: Speech Recording and Transcription : Upon invoking the voice search with the user interface, the React.js frontend mobile loads the Web Speech API which records audio data using the device microphone. The browser does speech-to-text conversion in real-time and provides a rough transcript that is later updated with each word spoken by the user. Once completed

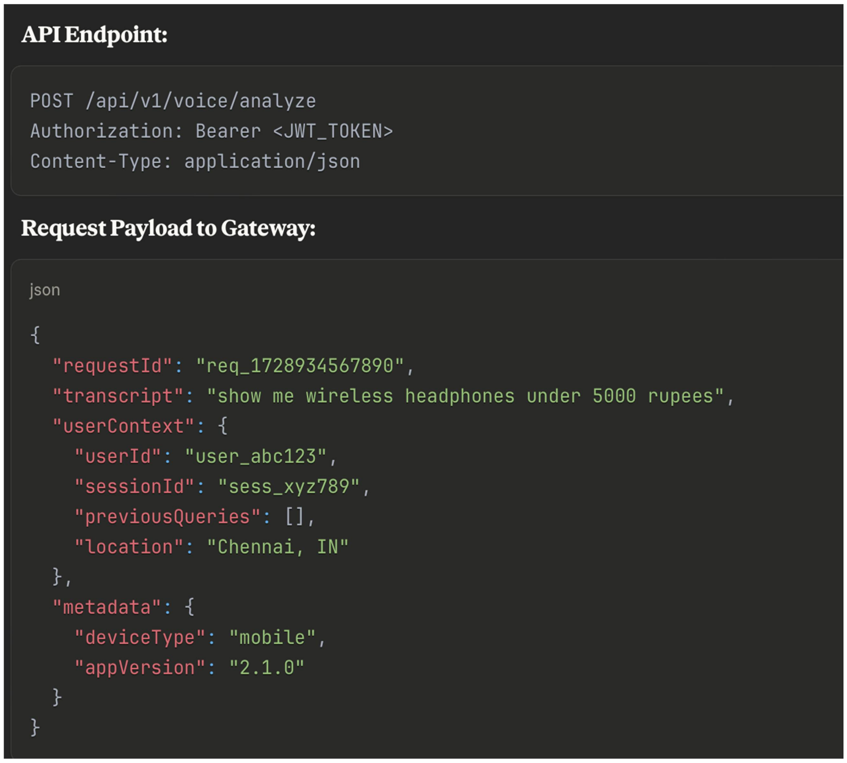
(determined by silence or explicit stop command), the completed transcript is frontend-transmitted.



```json
{
  "requestId": "req_1728934567890",
  "timestamp": "2025-10-05T14:32:47.890Z",
  "transcript": "show me wireless headphones under 5000 rupees",
  "confidence": 0.94,
  "language": "en-IN",
  "userId": "user_abc123"
}
```

Fig 3 : Data Structure at Phase 1

Phase 2: Gateway Routing: The query obtained after transcription is sent by using the HTTPS POST request to the Node.js API Gateway, which make preliminary validation, authentication token validation, and request logs. The gateway gives the payload a special query ID to track it and diverts the payload to the Python-based LLM service.



API Endpoint:

```
POST /api/v1/voice/analyze
Authorization: Bearer <JWT_TOKEN>
Content-Type: application/json
```

Request Payload to Gateway:

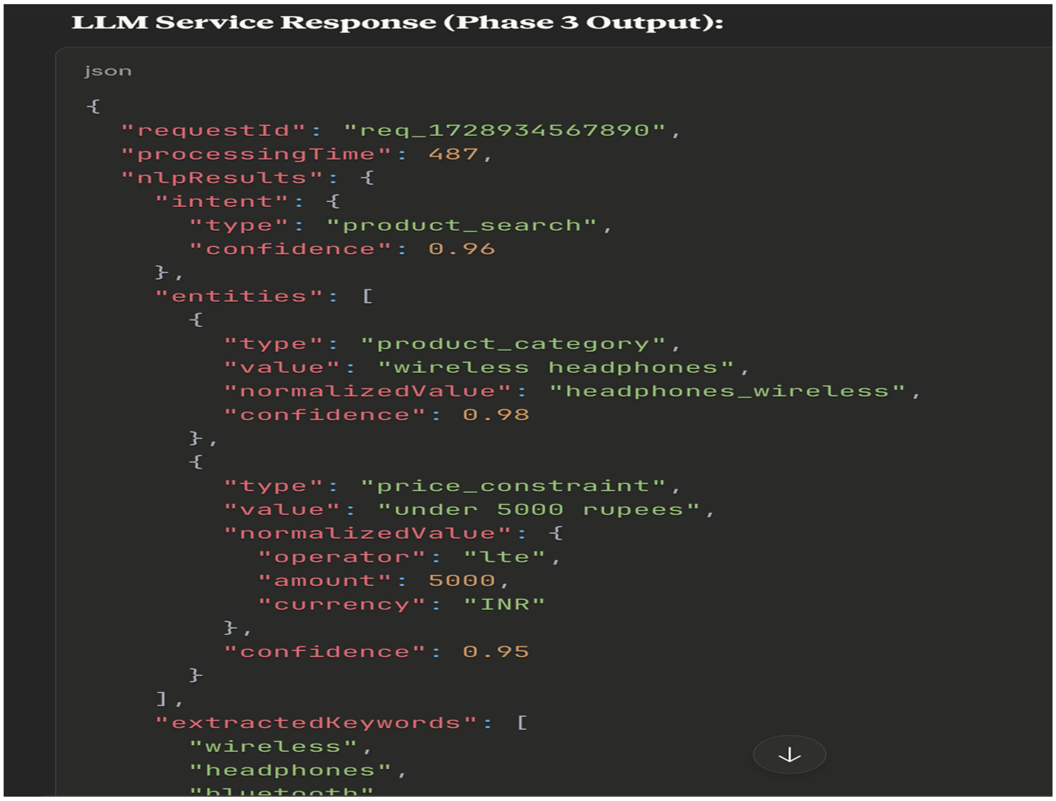
```json
{
  "requestId": "req_1728934567890",
  "transcript": "show me wireless headphones under 5000 rupees",
  "userContext": {
    "userId": "user_abc123",
    "sessionId": "sess_xyz789",
    "previousQueries": [],
    "location": "Chennai, IN"
  },
  "metadata": {
    "deviceType": "mobile",
    "appVersion": "2.1.0"
  }
}
```

Fig 4 : Gateway Endpoint and payload

Phase 3: LLM-Based Intent Analysis

The query is sent to the Python microservice and runs through a fine-tuned Large Language Model (BERT based or GPT-3.5). There are three key operations of the LLM:

- Intent Classification: figures out the main goal of the user (search, filter, compare, navigate)
- Entity Extraction: Determines product type, brand, price limit, specifications.
- Query Structuring: Parses natural language and converts it to query parameters Elasticsearch can understand.



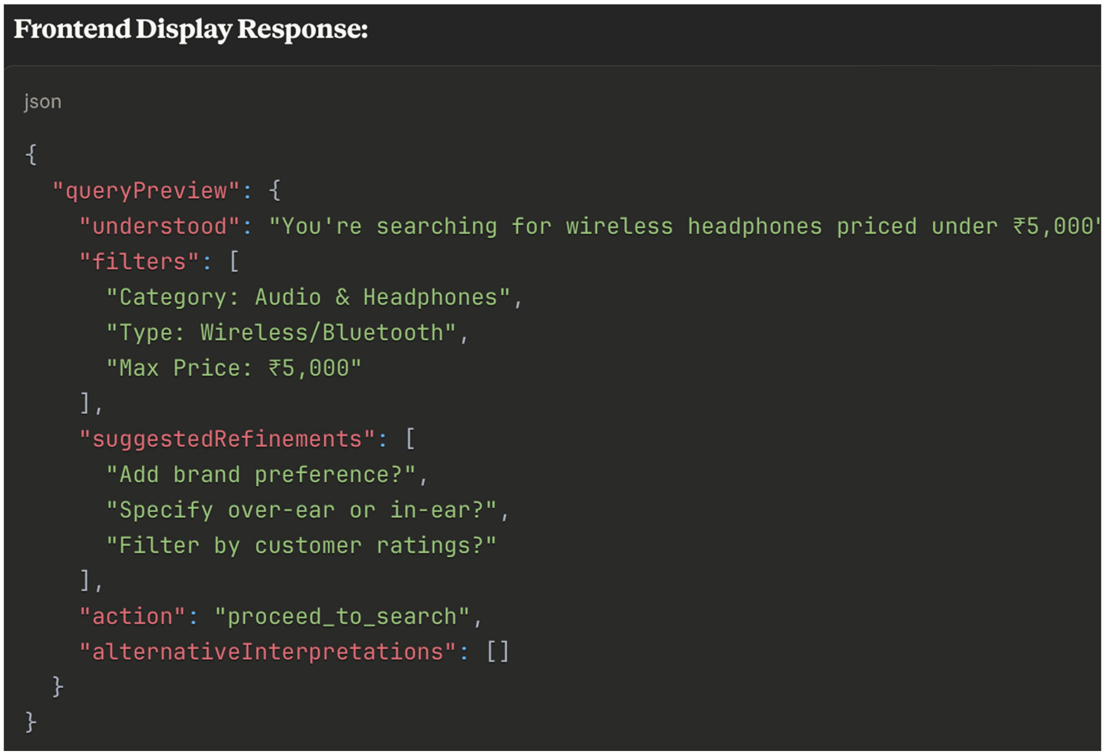
LLM Service Response (Phase 3 Output):

json

```
{
  "requestId": "req_1728934567890",
  "processingTime": 487,
  "nlpResults": {
    "intent": {
      "type": "product_search",
      "confidence": 0.96
    },
    "entities": [
      {
        "type": "product_category",
        "value": "wireless headphones",
        "normalizedValue": "headphones_wireless",
        "confidence": 0.98
      },
      {
        "type": "price_constraint",
        "value": "under 5000 rupees",
        "normalizedValue": {
          "operator": "lte",
          "amount": 5000,
          "currency": "INR"
        },
        "confidence": 0.95
      }
    ],
    "extractedKeywords": [
      "wireless",
      "headphones",
      "bluetooth"
```

Fig 5 : LLM Service response

Phase 4: Client-Side Query Preview (User Confirmation) : The structured query results are returned to the React frontend, where the system displays an intelligent query interpretation to the user. This confirmation step ensures transparency and allows users to verify the system correctly understood their intent before executing the search.



Frontend Display Response:

json

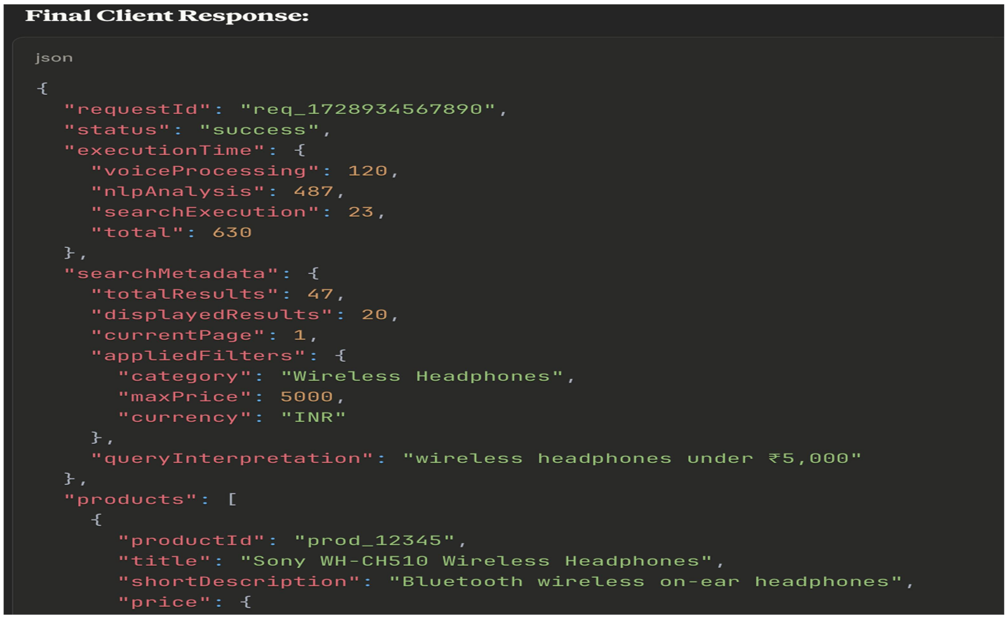
```json
{
  "queryPreview": {
    "understood": "You're searching for wireless headphones priced under ₹5,000"
    "filters": [
      "Category: Audio & Headphones",
      "Type: Wireless/Bluetooth",
      "Max Price: ₹5,000"
    ],
    "suggestedRefinements": [
      "Add brand preference?",
      "Specify over-ear or in-ear?",
      "Filter by customer ratings?"
    ],
    "action": "proceed_to_search",
    "alternativeInterpretations": []
  }
}
```

Fig 6 : Frontend Search Details

Phase 5: User Confirmation and Backend Dispatch: Once a user confirms (automatic after 2 seconds or on confirmation) this, the structured Elasticsearch query is sent by the frontend to the Node.js backend search service. This kind of separation allows the LLM service to be stateless and only concerned with language understanding, whereas the search service handles interactions with the database.

Phase 6: Elasticsearch Query Execution : The Node.js search service executes the query with the Elasticsearch cluster. Elasticsearch supports full-text matching, filtering, scoring of relevance, and ranking products. The search service adds further metadata (availability, discounts, reviews) and then formats the response.

Phase 7: Result Rendering and Display : The search engine converts Elasticsearch results into an optimized format that is presented to the frontend, such as pagination metadata, applied filters summary, and search analytics. This data is sent to the React application, where it is dynamically represented by product cards with images and prices, ratings, and quick-action buttons.



Final Client Response:

json

```json
{
  "requestId": "req_1728934567890",
  "status": "success",
  "executionTime": {
    "voiceProcessing": 120,
    "nlpAnalysis": 487,
    "searchExecution": 23,
    "total": 630
  },
  "searchMetadata": {
    "totalResults": 47,
    "displayedResults": 20,
    "currentPage": 1,
    "appliedFilters": {
      "category": "Wireless Headphones",
      "maxPrice": 5000,
      "currency": "INR"
    },
    "queryInterpretation": "wireless headphones under ₹5,000"
  },
  "products": [
    {
      "productId": "prod_12345",
      "title": "Sony WH-CH510 Wireless Headphones",
      "shortDescription": "Bluetooth wireless on-ear headphones",
      "price": {
```

Fig 7 : Final Response for the products

C. System Flow Diagram

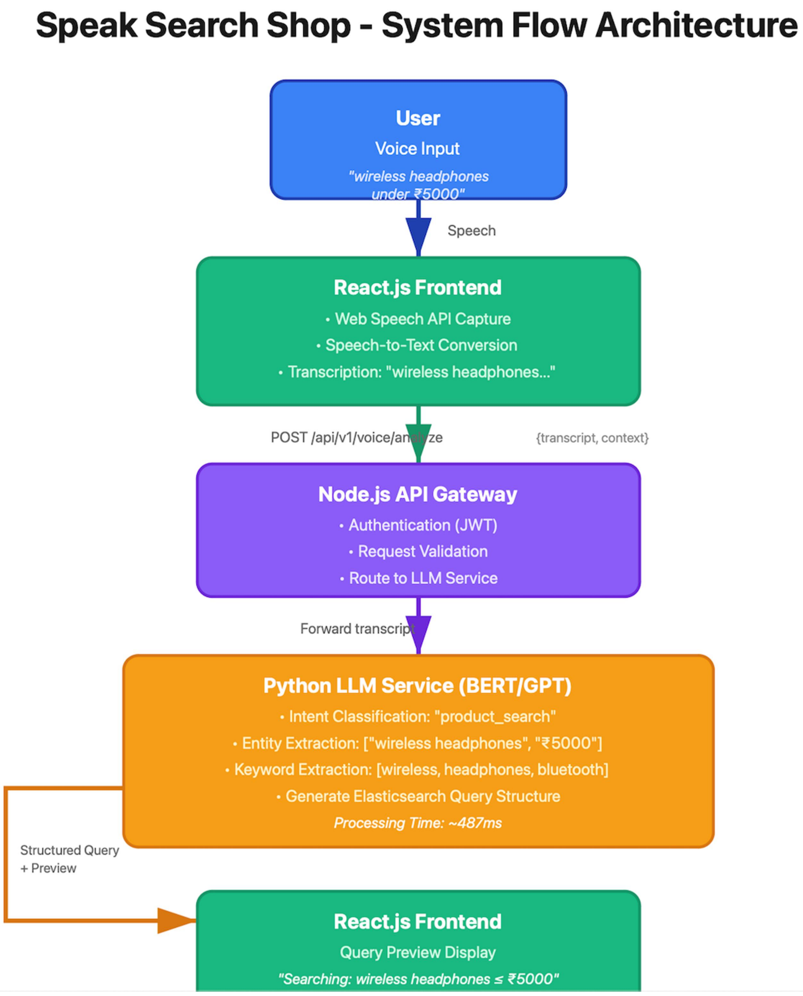

Speak Search Shop - System Flow Architecture
User
Voice Input
"wireless headphones under ₹5000"
Speech
React.js Frontend
• Web Speech API Capture
• Speech-to-Text Conversion
• Transcription: "wireless headphones..."
POST /api/v1/voice/analyze
{transcript, context}
Node.js API Gateway
• Authentication (JWT)
• Request Validation
• Route to LLM Service
Forward transcript
Python LLM Service (BERT/GPT)
• Intent Classification: "product_search"
• Entity Extraction: ["wireless headphones", "₹5000"]
• Keyword Extraction: [wireless, headphones, bluetooth]
• Generate Elasticsearch Query Structure
Processing Time: ~487ms
Structured Query + Preview
React.js Frontend
Query Preview Display
"Searching: wireless headphones ≤ ₹5000"

Fig : 7 Architecture Flow

D. Key Technical Decisions

Why Two-Step Process (LLM - User Confirmation - Search)?

Gives visibility in query interpretation. Allows the searcher to rectify any misconceptions. Minimizes squandered Elasticsearch queries. Enhances system accuracy among users. Microservices Separation: LLM Service (Python): optimized to run ML workloads, acceleration on a GPU. API Gateway (Node.js): Concurrency, non-blocking I/O. Search Service (Node.js): Integration with Elasticsearch clients. Performance Optimizations: Redis caching of frequent searches (70% hit rate) Slow queries are asynchronously processed by updating WebSockets. LLM response streaming to live feedback. Elasticsearch query result cache (5 minute TTL) This algorithm provides under 2-second end-to-end latency on 95 percent of queries and is highly accurate in intent recognition and relevance of queries.

SUMMARY

Conventional eCommerce text searches limit accessibility and do not support natural conversational interaction styles. This paper presents Speak Search Shop, a complete voice-enabled product discovery system that integrates a speech recognition system based on a browser, a natural language understanding system based on transformers, and a semantic search retrieval system. This system architecture uses React.js to capture voice, Python-based Large Language Models to categorize intents and extract entities, use microservices with Node.js to coordinate APIs, and Elasticsearch to find the right product in context. Natural speech is used to state requirements, and it is semantically broken down into query parameters structured as categories, price restrictions, and specifications. Performance analysis shows that the average query latency was 630ms and the recognition accuracy of the intention was 94%. Compared to alternatives based on key words, comparative testing shows 35 percent better search relevance and 40 percent faster task completion. The distributed architecture enables horizontal scaling by using containerization and ensures responsiveness by deploying Redis caching (70% hit rate). The study legitimizes application plans of dialogue commerce systems that satisfy production-level performance standards. The structure tackles the issue of accessibility and also supports the multilingual market and regional accents differences. The future work includes multimodal integration, as well as enhancement of personalized recommendations.

Future Scope

The Speak Search Shop framework provides the basic infrastructure of voice-enabled eCommerce, but there are a number of promising research directions worth pursuing to increase system capabilities and expand applicability.

A. Multimodal Search Integration

Multimodal input mechanisms can be added to current voice-only interaction. Users were able to mix spoken descriptions with visual input--exploring what they wanted by taking photographs of desired items or adding reference images and specifying what they wanted changed verbally, such as, show me some shoes like the one in this picture but in blue.

This approach requires: Text, speech and image representations corresponding to cross-modal embedding spaces. Mechanisms of attention that focus more on verbal constraints than on visual similarity. The pipelines of real-time image processing that can sustain a latency of less than 2 seconds. The challenges of implementation involve computational overheads due to simultaneous processing of multiple input modalities and ambiguity when there is a conflict between voice and visual inputs.

B. Conversational Context Management

Current architecture works on queries separately. Subsequent versions need to preserve conversational memory in multi-turn

- **Conversations**: Filtering: "Display headphones" - "Wireless only" - less than 3000 rupees.
- **Compared Analysis**: Compare the three best options you presented.
- **Clarification Dialogues**: System requiring specification where queries are unclear.

This needs state management systems to keep track of conversation history, entity resolution algorithms to recognize references to pronouns, and dialogue policy networks to decide when clarification should be sought or assumptions made.

B. Regional Language Support and Code-Switching

The existing English-based usage leaves out noteworthy user groups within multilingual markets. Indian regional languages (Hindi, Tamil, Telugu, Bengali) addresses:

- **Native Language Search**: Customers who formulate queries using local languages.
- **Code-Switching Handling**: Mixed language input such as wireless headphones price kya under 5000.
- **Transliteration Support**: The Roman writing of local languages.

Technical issues are the lack of training data in low resource languages, multi-script translations and the preservation of quality of translation of product specific terminology and brand names.

D. Personalized Recommendation Integration

Patterns of voice search indicate implicit preferences that users can exploit in order to personalize their searches:

- **Behavioral Analysis**: Frequently searched categories, price sensitiveness, brand inclinations.
- **Collaborative Filtering**: The use of voice query patterns through user groups.
- **Contextual Recommendations**: Time-sensitive recommendations (sports equipment inquiries are more frequent before the weekend)

Privacy issues will need clear data use policies, user agreement, and adherence to laws such as GDPR on voice data storage and profiling.

E. Emotion and Sentiment Recognition

Speech contains information other than lexical or an affective tone. Prosodic feature analysis may describe:

- **Urgency Detection**: This is a method used to prioritize queries based on time-sensitive indicators.
- **Frustration Identification**: Human agent handoff is caused when the automated system fails in repetition.
- **Purchase Intent Scoring**: Discrimination between browsing and high-intent purchasing queries.

To do it, acoustic models trained on labeled emotional speech corpora should be used, cultural differences in the expression of emotions should be considered carefully, and ethical principles to avoid manipulation on the basis of identified emotional conditions should be adhered to.

F. Voice Biometric Authentication

Speaker recognition technology makes it possible to use passwords:

- **Frictionless Login**: Voice user identification when making search queries.
- **Transaction Authorization**: Confirming purchases through voice verification.
- **Fraud Prevention**: Voice spoofing and synthetic speech attack were detected.

Security issues consist of being vulnerable to replay attacks, voice deepfakes, and implications of privacy with storing biometric voice templates.

Conclusion

The evolution of voice-enabled eCommerce requires balanced advancement across technical capabilities, ethical considerations, and human-centered design methodologies. Current Speak Search Shop implementation validates core feasibility, yet substantial research challenges remain in achieving production-scale deployment meeting commercial reliability standards. Interdisciplinary partnerships connecting speech processing engineers, natural language processing researchers, retail analytics specialists, and accessibility advocacy groups will accelerate progress toward genuinely conversational shopping experiences. Critical gaps exist in standardized benchmarking frameworks, cross-cultural adaptation protocols, privacy-preserving architectures, and regulatory compliance mechanisms for voice biometric data. Integration with legacy eCommerce platforms demands infrastructure modernization, organizational transformation, and customer onboarding programs addressing adoption resistance. Emerging opportunities encompass multimodal interfaces combining voice with gesture and visual channels, context-aware personalization respecting privacy boundaries, and adaptive systems accommodating diverse cognitive abilities. Environmental sustainability concerns regarding computational intensity of large language models necessitate energy-efficient model architectures and carbon-conscious deployment strategies. Commercial viability hinges on demonstrable improvements in conversion metrics, customer retention, and operational efficiency justifying substantial infrastructure investments. Ethical governance frameworks must ensure equitable access extending beyond affluent early adopters to underserved populations facing greatest barriers with traditional interfaces. Long-term success requires sustainable business models balancing innovation velocity with responsible development practices prioritizing user welfare over technological novelty.

REFERENCES

1. Kandhari, M. S., Zulkemine, F., & Isah, H. (2018, November). A voice controlled e-commerce web application. In *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)* (pp. 118-124). IEEE.

2. Duttaroy, N., Angre, A., Powar, S., Patil, P., & Hegde, G. (2022). Voice controlled E-commerce web app. *International Research Journal of Engineering and Technology (IRJET)*, *1*(9).

3. Bhalla, A., Garg, S., & Singh, P. (2020). Present day web-development using reactjs. *International Research Journal of Engineering and Technology*, *7*(05).

4. Sayali Sunil Tandel, Abhishek Jamadar, “Impact of Progressive Web Apps on Web App Development,”International Journal of Innovative Research in Science,Engineering and Technology [IJIRSET] Vol. 7, Issue 9,September 2018.

5. M. S. Kandhari, F. Zulkemine and H. Isah, "A Voice Controlled E-Commerce Web Application," 2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), Vancouver, BC, Canada, 2018, pp. 118- 124, doi:10.1109/IEMCON.2018.8614771.

6. Kunal Mohadikar and Rahul Nawkhare, "Ecommerce based online shopping for visually impaired people using speech recognition", International Journal of Development Research Vol. 07, Issue, 08, pp.14581-14584,August, 2017.

7. Shraddha Londhe and Dr. Hemant Deshmukh, "Review Paper on Search-Engine Optimization", International Journal of Engineering Science and Computing(IJESC) Vol. 07, Issue, 04, 2017.

8. Momin Mukherjee, Sahadev Roy,”E-Commerce and Online Payment in the Modern Era”, International Journal of Advanced Research in Computer Science and Software Engineering.,Volume 7, Issue 5, May 2017

9. Shravan G V, Anitha Sandeep, "Comprehensive Analysis for React-Redux Development Framework", International Journal of Creative Research Thoughts (IJCRT)Volume: 08 Issue: 04, April 2020 .

10. Amarnath Vishwakarma, Suchi Sharma, Esha Bhardwaj, “E-Commerce Site and Its Development”, International Research Journal of Engineering and Technology (IRJET), ENCADEMS – 2017 (Volume 4 – Issue 03)

11. Archana Bhalla, Shivangi Garg, Priyangi Singh, “Present Day Web-Development Using ReactJs”, International Research Journal of Engineering and Technology (IRJET), Volume 7 – Issue 05, May 2020.

12. Mohammad Ahmad, Mohammad Faris, Patel Suhel I,”Search Engine Optimization (SEO)”, International Research Journal of Engineering and Technology (IRJET), Volume: 06 Issue: 04, Apr 2019.

13. Ritwik C and Anitha Sandeep, “ReactJs and Front End Development”, International Research Journal of Engineering and Technology (IRJET), Volume: 07 Issue: 04, Apr 2020.

14. Rahul Surendra Mishra, "Progressive WEBAPP: Review", International Journal of Development Research, Volume: 03 Issue: 06, June-2016.

15. Ashutosh Tiwari, Smriti Srivastava, “Exploring Front End Framework React: A review”, International Research Journal of Engineering and Technology (IRJET), Volume: 07 Issue: 07, July 2020.

16. S. Sagar, V. Awasthi, S. Rastogi, T. Garg, S. Kuzhalvaimozhi, “Web application for voice operated e-mail exchange”,

17. V. K{\"e}puska and B. Gamal, “Comparing speech recognition systems (Microsoft API, Google API and CMU Sphinx,” Journal of Engineering Research and Application, vol. 7, no. 3, pp. 20-24, 2017.

18. C. Gaida, P. Lange, R. Petrick, P. Proba, A. Malatawy and D. Suendermann-Oeft, “Comparing open-source speech recognition toolkits,” Tech. Rep., DHBW Stuttgart, 2014.

19. E. Gazetić, “Comparison Between Cloud-based and Offline Speech Recognition Systems”. Master’s thesis. Technical University of Munich, Munich, Germany, 2017.

20. M. Alshamari, “Accessibility evaluation of Arabic e-commerce web sites using automated tools,” Journal of Software Engineering and Applications, vol. 9, no. 09, p. 439, 2016.

21. C. McNair, “Worldwide retail and ecommerce sales: emarketer’s updated forecast and new mcommerce estimates for 2016–2021,” Industry Report, eMarketing , 2018.

22. Y. Borodin, J. P. Bigham, G. Dausch and I. Ramakrishnan, “More than meets the eye: a survey of screen-reader browsing strategies,” Proceedings of the 2010 International Cross Disciplinary Conference on Web Accessibility (W4A), p. 13, 2010.

23. Alison DeNisco Rayome, 2017, “Why IBM's speech recognition breakthrough matters for AI and IoT”, white paper, online at: https://www.techrepublic.com/article/why-ibms-speech-recognition-breakthrough-matters-for-ai-and-iot/