INTERNATIONAL STANDARD

ISO/IEC 24661

First edition
2023-05

Information technology — User interfaces — Full duplex speech interaction

Technologies de l'information — Interfaces utilisateur — Interaction vocale en duplex intégral

Reference number
ISO/IEC 24661:2023(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives or www.iec.ch/members_experts/refdocs).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents) or the IEC list of patent declarations received (see https://patents.iec.ch).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 35, *User interfaces*.

Any feedback or questions on this document should be directed to the user’s national standards body. A complete listing of these bodies can be found at www.iso.org/members.html and www.iec.ch/national-committees.

Introduction

Speech interaction user interface (UI) has been widely used for industrial applications and daily services. For example, it can be applied to automatic customer service in the telecommunication industry as a part of an interactive voice response system. From a communication point of view, a speech interaction UI can be recognized as a duplex-based system which enables bidirectional communication. In the early stages, speech interaction UIs for conventional dialogue systems were generally half duplex (HDX) based and were designed to be in a turn-oriented work mode. As the requirements of human-machine interaction have grown in complexity and diversity, the turn-oriented speech interaction UI has become unfit for a conversation between humans and machines.

Currently, full duplex (FDX) techniques are used in the speech interaction UI to support session-oriented conversations between humans and machines. The most significant differences between turn-oriented and session-oriented speech interactions are continuity and naturalness, which have made great progress in various applications of speech interaction UI, e.g. smart speaker, chatbot, intelligent assistant.

In recent years, a growing number of FDX speech interaction UIs have been studied and developed. This requires a common understanding of general models and specifications through standardization activities. In response to the standardization needs both from industry and academia, this document intends to provide a reference architecture, functional components and technical requirements of FDX speech interaction UI. For the benefit of system designers, developers, service providers and ultimate users, this document is composed of the following clauses:

— Clause 5 describes a functional view and general features of FDX speech interaction;

— Clause 6 provides a reference architecture and functional layers of FDX speech interaction UI;

— Clause 7 specifies the functional requirements regarding each functional layer;

— Clause 8 discusses the processes of FDX speech interaction UI;

— Clause 9 describes security and privacy considerations related to FDX speech interaction UI.

Information technology — User interfaces — Full duplex speech interaction

1 Scope

This document specifies user interfaces (UIs) designed for full duplex (FDX) speech interaction. It also specifies the FDX speech interaction model, features, functional components and requirements, thus providing a framework to support natural conversational interfaces between humans and machines. It also provides privacy considerations for applying FDX speech interaction.

This document is applicable to UIs for speech interaction and communication protocols for setting up a session-oriented FDX interaction between humans and machines.

This document does not define the speech interaction engines themselves or specify the details of specific engines, devices and approaches.

2 Normative references

There are no normative references in this document.

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at https://www.electropedia.org/

3.1
duplex
method of communication capable of transmitting data in both directions

[SOURCE: ISO 21007-1:2005, 2.18]

3.2
full duplex
FDX
method of communication capable of transmitting data in both directions at the same time

[SOURCE: ISO 21007-1:2005, 2.25]

3.3
functional unit
entity of hardware or software, or both, capable of accomplishing a specified purpose

Note 1 to entry: Functional units can be integrated as a system.

[SOURCE: ISO/IEC 2382:2015, 2123022, modified — Note 1 to entry has been changed and Note 2 and 3 to entry have been removed.]

3.4
half duplex
HDX
method of communication capable of transmitting data in both directions but only in one direction at any time

[SOURCE: ISO 21007-1:2005, 2.27]

3.5
microphone array
system that is composed of multiple microphones with definite spatial topology, which samples and filters the spatial characteristics of signals

3.6
speech interaction
activities of information transmission and communication between humans and a system through speech

Note 1 to entry: A system can be seen as a combination of *functional units* (3.3).

3.7
speech recognition
automatic speech recognition
ASR
conversion, by a *functional unit* (3.3), of a speech signal to a representation of the content of the speech

Note 1 to entry: The content to be recognized can be expressed as a proper sequence of words or phonemes.

[SOURCE: ISO/IEC 2382:2015, 2120735, modified — Notes 2 to 4 to entry have been removed.]

3.8
speech synthesis
generation of speech from data through a mechanical method or electronic method

Note 1 to entry: Speech can be generated from text, image, video and audio. The process of conversion from text to speech is the main approach in *speech interaction* (3.6).

Note 2 to entry: The result of speech synthesis is also called "artificial speech" in order to differ from natural speech through human vocal organs.

3.9
voice activity detection
VAD
process of analysis and identification of the starting and ending points of valid speech in a continuous speech stream

3.10
voice trigger
process in a system in the audio stream monitoring state, which switches to command word recognition, continuous speech recognition and other processing states after the detection of certain features or events

4 Symbols and abbreviated terms

AAC	advanced audio coding
AC3	audio coding 3
AI	artificial intelligence
ASR	automatic speech recognition
EVRC	enhanced variable rate codec
FDX	full duplex
HDX	half duplex
ML	machine learning
MP3	MPEG audio layer 3
NER	named entity recognition
NLG	natural language generation
NLP	natural language processing
NLU	natural language understanding
SNR	signal-to-noise ratio
TTS	text-to-speech
UI	user interface
VAD	voice activity detection
WAV	waveform audio file format
WMA	Windows media audio

5 Overview of FDX speech interaction UI

5.1 Functional view

Speech interaction UI can function as a communication channel between a human and a system. A user can apply a speech interaction UI to have a conversation with a system, while a system can also respond to the user with synthesized speech through the speech interaction UI. Such bidirectional communication can be viewed as a duplex speech interaction. With different data transmission sequences, there are two types of duplex speech interactions, including HDX mode and FDX mode.

In the case of HDX speech interaction, both a human and a system can communicate with each other in one direction at a time. An HDX speech interaction is characterized as a turn-oriented dialogue, where a system will return to the default state after it finishes one round of dialogue. In addition, the system cannot collect speech signals during the process of its speech broadcasting.

NOTE 1 A typical HDX-based communication system is a two-way radio such as walkie-talkie. A walkie-talkie uses a "push-to-talk" button to control the signal transmission channel. A user can turn on the transmitter and turn off the receiver by using the button, so that the voice from remote users cannot be heard.

In contrast to HDX speech interaction, FDX speech interaction allows a human and a system to communicate with each other simultaneously. An FDX speech interaction is characterized as a session-oriented conversation, where a system keeps the conversation continuous and ensures that both user and system are in the same context after two or more rounds of dialogue. In addition, the user and the system can speak within the same interval of time. Example scenarios of FDX speech interaction are shown in Annex A.

NOTE 2 A typical FDX-based communication system is the telephone, where both local and remote users can speak and be heard at the same time.

From the functional point of view, a system can keep receiving the input data from the user and providing feedback to them through an FDX speech interaction UI during the whole human-machine conversation. Figure 1 depicts a functional view of FDX speech interaction UI that includes inputs, processing and outputs.

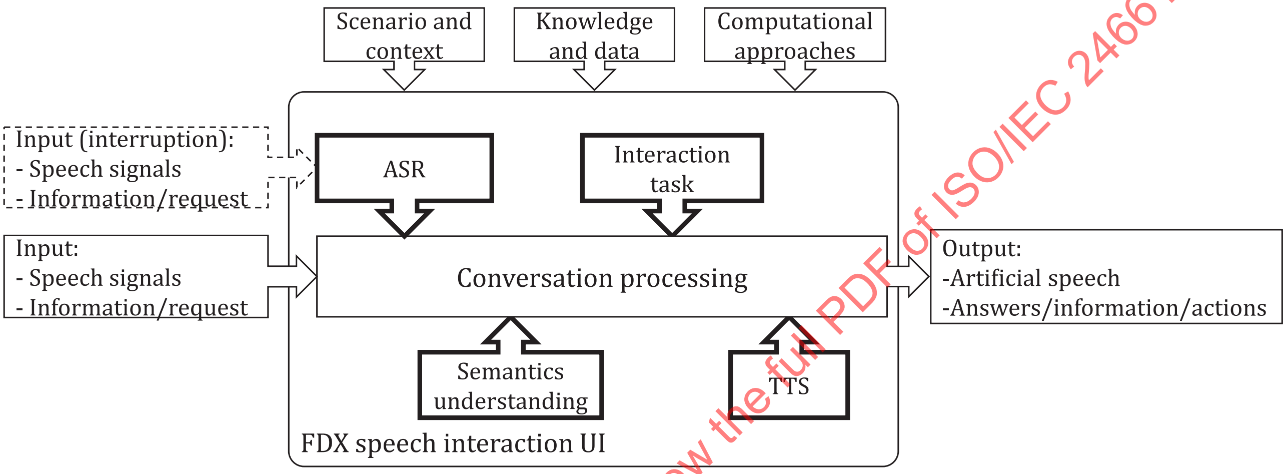


Figure 1 — Functional view of FDX speech interaction UI

This functional view provides a non-technical description of how an FDX speech interaction uses UI to achieve its goal. Through the FDX speech interaction UI, a system can receive the input speech signals, transcript the useful signals into the text, abstract the semantic information from transcription text, make predictions and decisions regarding interaction tasks based on semantic information, and either take actions based on the decisions or provide speech feedback to users as the outputs, or both. In contrast to HDX mode, an FDX speech interaction is characterized by functions of continuous speech acquisition by a system after it has been awoken once. Such function can be performed even when a system is outputting synthesized speeches or other actions. This is considered to be a conversational interruption, i.e. technically, both uplink speech stream and downlink speech stream may take place at the same time. An FDX speech interaction UI shall have abilities to execute the conversation processing whenever there are speech interruptions and to generate updated outputs based on the new inputs.

During this process, scenarios and contexts can be used to define the semantic range of the conversation. A conversation can be cross scenarios and contexts. General knowledge and big data are required for the conversation processing. Computational approaches such as cloud computing and AI computational approaches should be introduced in the FDX speech interaction UI. Such functional components are applied to performing intelligent conversation processing, which is a distinguishing characteristic of FDX mode compared with HDX mode

5.2 Main characteristics

5.2.1 General

To demonstrate the breadth of FDX speech interaction UI, some common characteristics are described in 5.2.2 to 5.2.8. In the aggregate, these characteristics are intrinsic to many FDX speech interaction UIs, which will differentiate FDX speech interaction UIs from non-FDX speech interaction UIs. The list of characteristics of FDX speech interaction UIs is not exhaustive, but broadly conceptual and not tied to a specific methodology or architecture.

5.2.2 Continuous

Through an FDX speech interaction UI, a user can keep talking as continuous inputs, while the system can keep receiving and processing the input data.

5.2.3 Natural

An FDX speech interaction UI can support a natural conversation between a human and a system. A system only needs to be awoken once at the beginning of the conversation. A user can talk at will and freely interrupt the system at any time during a conversation.

5.2.4 Adaptable

An FDX speech interaction UI can adapt to different changes in itself and the environment in which it is deployed. It can be used in different vertical industries and applied to cross-domain applications and tasks by feeding on dynamic data and updating status based on new data.

5.2.5 Initiative

An FDX speech interaction UI can exhibit dynamic predictions of conversational intention based on external data sources, control the pace of conversation, and actively provide feedback to guide the user for further steps.

5.2.6 Context-based

An FDX speech interaction UI builds its core functions on context, e.g. semantic understanding, historical information inheritance, data analysis and dialogue generation.

5.2.7 Knowledge-based

An FDX speech interaction UI can use knowledge from multiple sourced information, including contextual information, historical information, retrieval information and user information. This information can be stored in the general knowledge and database.

NOTE Retrieval information refers to information that is searched from other resources, e.g. internet website, database and knowledge base.

5.2.8 Model-based

An FDX speech interaction UI operates with various degrees of utilization of an acoustic model and language model. With the rapid development of emerging technologies, some FDX speech interaction UI are also embedded with cloud frameworks and AI-related models, e.g. convolutional neural network (CNN), recurrent neural network (RNN) and long short-term memory (LSTM) network.

6 Reference architecture of FDX speech interaction UI

6.1 General

Based on the functional view described in Figure 1, a reference architecture of FDX speech interaction UI is represented in terms of functional layers depicted in Figure 2. It provides a common understanding of function units and their relationships, which are technically necessary to construct an FDX speech interaction UI. While this reference architecture is not limited to a specific base technology (e.g. FDX speech interaction UI built with neural networks), it does not encompass every type of dynamic FDX speech interaction UI.

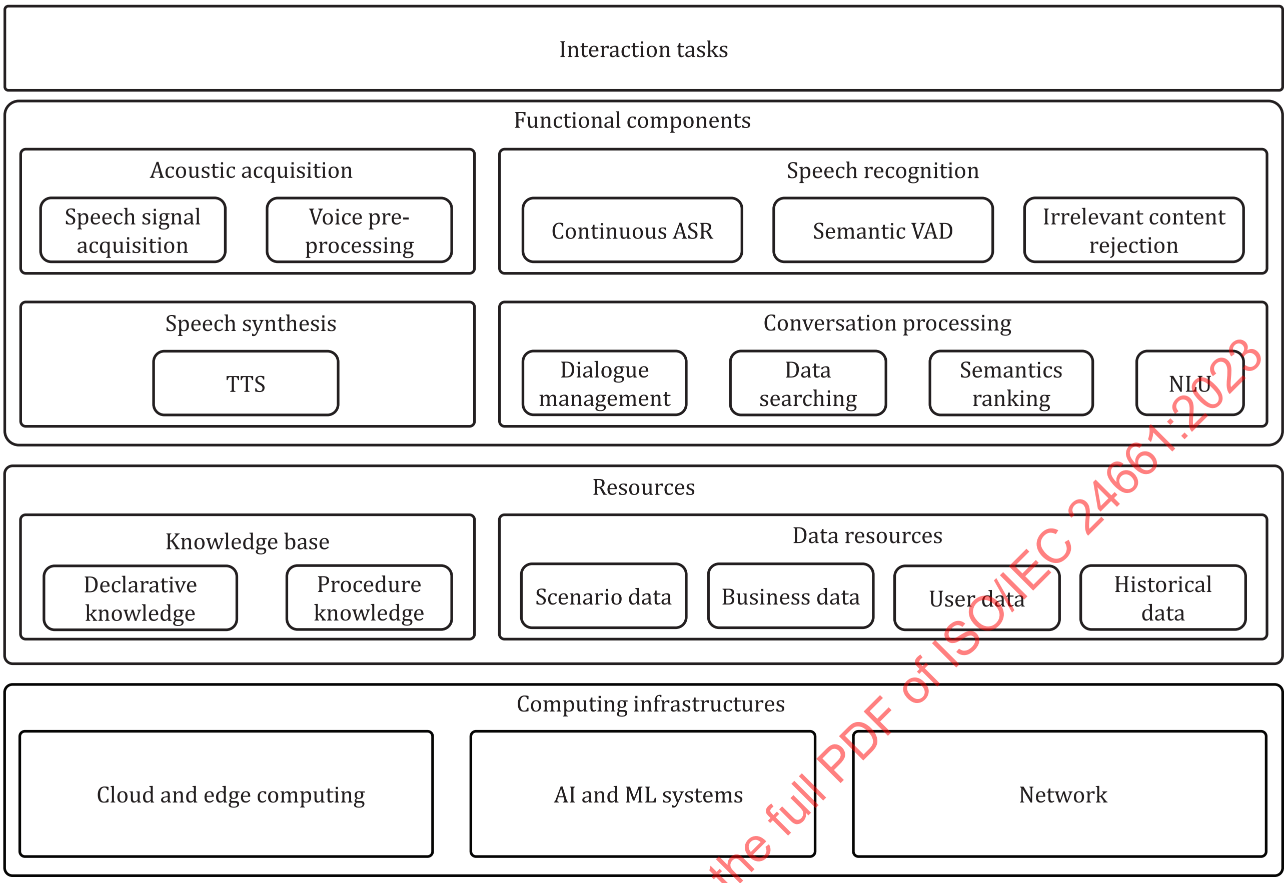


Figure 2 — Reference architecture of FDX speech interaction UI

This reference architecture consists of multiple layers and components. Such layers can be described in terms of the inputs, the outputs and the intents or functions. Each layer and its components can be used and tested separately. All layers can be integrated together to enable users to have conversations with the system and help to fulfil their requirements.

NOTE The system can be various smart devices, e.g. smart phone, smart home appliance, intelligent assistant app and customer service robot.

Speech data streams are transmitted through two physical channels. The upstream channel transmits speech data from the user to system. The downstream channel transmits speech data from the system to the user. Both channels shall be able to work at the same time without mutual-interference, thus to provide the system with the capability of “hearing” while “talking”.

6.2 Interaction tasks

Interaction tasks refer to some specific purposes and requirements that need to be satisfied using an FDX speech interaction UI. One or more tasks can be defined for FDX speech interaction UI.

Each interaction task can be logically designed using traditional software engineering approaches, which involves defining the scenarios, the environmental features, the input and output, the function units, the database and the data flow.

Interaction tasks differ in the types of scenarios and the user requirements. Examples of interaction tasks can include, e.g. phone call, navigation, home service, chatting. While the scenarios should be defined in a general design, methods to resolve the specific problems should be addressed during the construction process. For example, using FDX speech interaction for a task of navigation, while a car driving scenario should be defined in the top-level design process, the point of interest (POI), the key words and the statement of specific enquiry of road and route should be addressed during the construction process.

6.3 Functional components

6.3.1 General

In FDX speech interaction UI, interaction interface is a functional combination of “hearing”, “recognizing”, “understanding” and “talking”. An interaction interface is created using functional components including acoustic processing, speech recognition, conversation processing and speech synthesis. Figure 3 shows an example procedural relationship of the main functional components referring to the input and the output of FDX speech interaction.

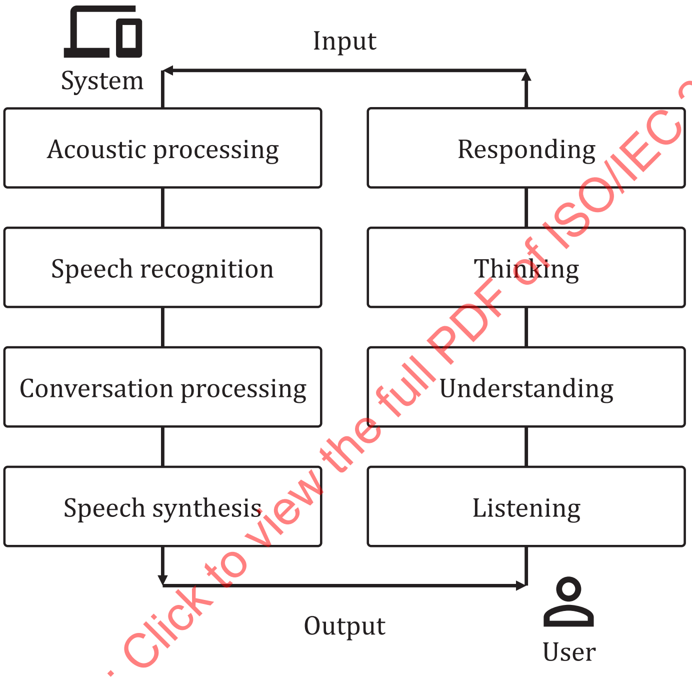


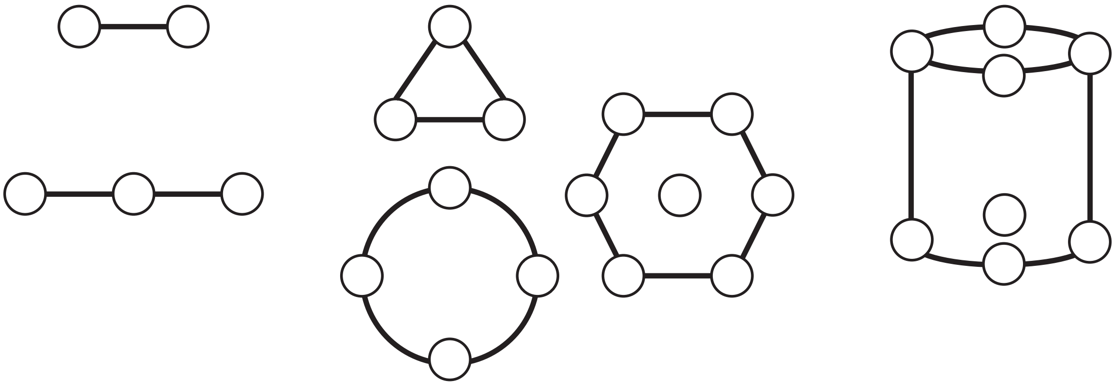
Figure 3 — Example procedural relationship of the main functional components of FDX speech interaction UI

While the components described in the 6.3.2 to 6.3.5 play core functions in an FDX speech interaction UI, additional functions can be added or modified to fulfil the user’s extra requirements. Although there are temporal relations in the process of speech interaction between these components, some components can interact with others at the same time. For example, the semantic voice activity detection (VAD) and the irrelevant content rejection in speech recognition also use the NLU and the semantics processing in conversation processing. The voice trigger function based on ASR is mainly used in the acoustic front-end.

6.3.2 Acoustic acquisition

Acoustic acquisition refers to functions of speech signal acquisition and voice pre-processing using microphone or microphone array in the front-end of FDX speech interaction UI.

A microphone array is generally composed of two or more microphones in the linear form, the planar form and the spatial steric form. Front-end related algorithms (e.g. beamforming algorithm, de-noise algorithm) should also be included as a part of microphone array. Examples of the structure of a microphone array are shown in Figure 4.



a) Linear form **b) Planar form** **c) Spatial steric form**

Figure 4 — Examples of microphone array structure

A microphone array is often used to take sound samples and to process the spatial characteristics of the acoustic field. In FDX speech interaction UI, microphone array is not only used to collect speech signals, it also can be used for different voice pre-processing functions. Unlike HDX speech interaction, an FDX speech interaction is featured by capturing the speech signals from the target speaker correctly and precisely, often in a complex environment. Such an environment is filled with various sounds including background noises or echo sounds or speech voices from other people. Most importantly, in order to perform a conversational interruption function, it shall be able to distinguish the speaker's speech from the output synthesized speech generated by the system itself. Therefore, to enable an FDX speech interaction, the input speech signals need to be pre-processed. The following non-exhaustive list describes some typical functions that can be applied:

a) Speech enhancement: the process of extracting pure speech signals from a noisy background, especially in a complex acoustic environment, when the speech signals have interference from, or are even submerged by, all kinds of noise (including background noise and unrelated voices from other people). The beam forming approach can be used to restrain noise and enhance speech.

b) Acoustic source localization: the use of the microphone array to calculate the angle and distance of the target speaker, in order to implement speaker tracking and choosing the speech direction. A speaker does not need to move the microphone to adjust its receiving direction and has high mobility.

c) Dereverberation: reverberation often refers to the acoustic phenomenon that, when sounds are propagated in an enclosed space (e.g. indoor space), the waves will be reflected by walls, ceiling, floor and other obstacles forming a superposition with the original sound. Due to reverberation, the asynchronous speech signals will overlap each other, resulting in the masking effect. A microphone array can use the following approaches to implement dereverberation:

   1) blind signal enhancement approach taking the reverberation as the common additive noise and applying a speech enhancement algorithm to them;

   2) beam forming based approach forming a voice pickup beam in the target direction by weighted summation of collected signals and attenuating the reflected sound from other directions;

   3) inverse filtering approach using microphone array to estimate the room impulse response (RIP) and using a reconstruction filter to compensate the dereverberation.

d) Speech source extraction and separation: the process of extracting the target speech signal from multiple sound signals and speech source separation that is intended to extract the multiple mixed speech signals. Both the beam forming approach and the blind source separation approach, including the principal component analysis and independent component analysis, can be used for this function.

e) Acoustic echo cancellation (AEC): it ensures the UI can collect speech signals when it is broadcasting audio functions (e.g. music, artificial speech) and it plays an important role in the FDX speech interaction UI. When a user at the far-end A is speaking, the speech is collected by the microphone and will be transmitted to the communication device at the near-end B and broadcasted by a speaker. The speech signals will then be picked up by the microphone at the near-end B, forming an acoustic echo. Such echo signals will return to the far-end A through transmission and be broadcasted through a speaker at the far-end A. The user will then hear his/her own voice. The acoustic echo signal can largely impact the speech acquisition effect, and therefore shall be removed during the speech collection process. An adaptive filter with a finite impulse response (FIR) structure can be used for the AEC function.

6.3.3 Speech recognition

6.3.3.1 General

In FDX speech interaction UI, a speech recognition component is used to convert speech signals into texts, which represent the content of the speech. It consists of continuous ASR, semantic VAD and irrelevant content rejection.

6.3.3.2 Continuous ASR

A continuous ASR unit attempts to recognize the continuous speech stream. It is composed of encoder, acoustic model, language model, lexicon and decoder. Figure 5 shows an example of a continuous ASR framework.

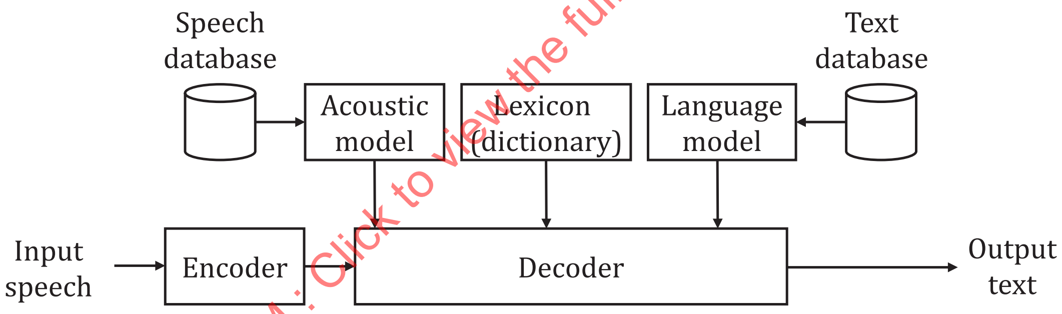


Figure 5 — Example framework of continuous ASR

Encoder refers to an extracting feature from speech signals. It can be used to transform each wave frame into a multi-dimensional vector that represents the utterance information. Prevalent features in continuous ASR include linear prediction coefficients (LPCs), perceptual linear predictive (PLP), tandem, bottleneck, filterbank, linear predictive cepstral coefficient (LPCC) and Mel-scale frequency cepstral coefficients (MFCCs).

NOTE Tandem and bottleneck can be extracted using a neural network. Specifically, tandem features are obtained by reducing the dimension of a posterior probability vector of the corresponding class of nodes in the output layer of neural network and splicing with MFCC or PLP features.

The acoustic model training is usually implemented on feature vector and output phoneme information. The lexicon refers to word and phoneme correspondence, e.g. correspondence between phonetic alphabet/symbol and characters/word. The language model is used to obtain the probability of word correlation by the model training on a large amount of text information. The decoder algorithm is used to output texts from feature extracted speech data through the acoustic model, lexicon and language model.

Depending on ASR, the function of voice trigger plays the role of initiating a conversation between a human and a system. Voice trigger often refers to the wake-up of the system using voice command words or phrases through FDX speech interaction UI. The voice trigger often functions at the front-end. In order to provide a natural conversation, the command words can be combined with continuous speech for the voice trigger, which is introduced as a “one-shot” voice trigger.

6.3.3.3 Semantic VAD

The purpose of semantic VAD is to identify and eliminate the silent period from a speech signal stream and distinguish between speech and non-speech based on time-frequency domain features and semantic features. Conventional acoustic VAD approaches, including the energy-based approach, periodic feature-based approach and multi-feature fusion approach and zero-crossing rate, can be used for near-field ASR.

A continuous speech stream often contains a variety of background noise and is affected by the speech speed and way of speaking. The acoustic VAD based on the energy or zero-crossing rate method is not effective. Considering many scenarios with high noise (i.e. low SNR) and far-field speech pickup, the semantic VAD method should be used. The ML method (e.g. LSTM, deep neural network) is used to calculate the semantics truncation probability to dynamically set the silence waiting time and output the sub-sentence text.

6.3.3.4 Irrelevant content rejection

The purpose of irrelevant content rejection is to check whether the FDX speech interaction UI can distinguish and reject the input content that cannot be processed or should not be processed. Such input content is generally unrelated to the interaction task as well as the conversation topic or context. It can also include invalid speech. More importantly, disambiguation can be achieved through scenario semantics rejection.

6.3.4 Conversation processing

6.3.4.1 General

Conversation processing is used for system “understanding” and “thinking” purposes. Understanding refers to the NLU and semantics ranking functions. Thinking refers to the data searching and dialogue management functions. Conversation processing can also be regarded as a process of dialogue jumping.

6.3.4.2 NLU

Generally, NLU refers to extraction of information from text or speech communicated to it in a natural language, and the production of a description for both the given text or speech, and what it represents. NLU can be seen as a part of NLP, which will convert text or speech into an internal description which is supposed to be the semantic representation of the input.

Two fundamental functions in NLU are used to support FDX speech interaction UI including NER and intention understanding. The purpose of NER is to seek to recognize and label the denotational names of, e.g. person, location, organization. Based on NER, the function of intention understanding can include domain classification, intention recognition and semantic labelling. Figure 6 depicts the relations among domain classification, intention recognition and semantic labelling and their roles in the levels of intention understanding.

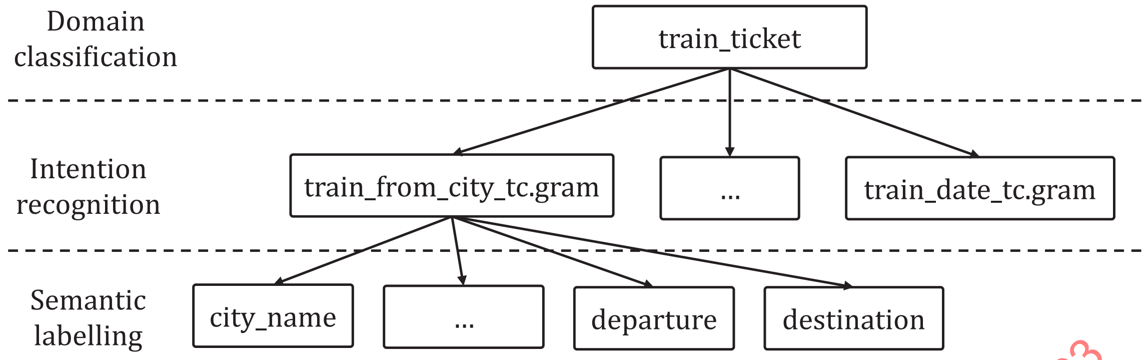


Figure 6 — Relations among domain classification, intention recognition and semantic labelling

The top-level is domain classification, which involves classifying the meaning of a sentence into a high-level domain category. The middle level is intention recognition, which is to recognize more details of the statement with the grammar network and map them to a defined expression base in the form of augmented Backus-Naur form. The bottom level is semantic labelling, also called attribute extraction, which refers to a process of generating and tagging labels representing the specific concepts or meanings (e.g. NER results) to the key word or the statement with semantics slots. The semantic labelling can also be regarded as a sequence labelling task for selecting useful semantic meaning of a speaker's intention, which can be solved using rule-based or ML-based approaches.

6.3.4.3 Semantics ranking

After the NLU, it is available for generating one or more semantics paths. The intention is to use semantics ranking to find the best semantics result. Generally, the semantic paths outputted by NLU are disordered so it should use the rule-based, the grammar-based and the model-based approaches to determine an optimum path to define the final semantics.

6.3.4.4 Data searching

Data searching can be used to convert a user's semantic information into a business request, find the data that meet the user's needs from a large amount of business data, and return it to the user and output response text information for NLG according to the searching results.

Data retrieval can include semantics inheritance, semantics post-processing, information source search, semantic correction and business data sorting. Data retrieval should be deployed in the form of cloud services, to quickly and accurately meet the business needs of users based on strong computing capability.

6.3.4.5 Dialogue management

In FDX speech interaction UI, dialogue management refers to an integrate function of NLG, dialogue guidance and tempo control. NLG can be further partitioned into six mutually exclusive tasks:

a) text content determination: task of deciding what information can be included in the text;

b) text structure: task of determining the order in which information is presented in the text;

c) sentence aggregation: task of deciding what information to present in a single sentence;

d) lexicalization: task of finding the right word or phrase to express the information;

e) referring expression generation: task of selecting words and phrases to identify the domain object;

f) linguistic realization: task of combining all the words and phrases into well-formed sentences.

Dialogue guidance refers to updating the current user's scenario and status by using the scenario state semantics and the searched data, and generating the prompts based on those semantic data, to guide the dialogue or open a new topic.

Tempo control refers to coordination and control of the conversation tempo according to the scenario data, speaker status (speaker type, emotion) and conversation state (speech speed, intonation), so as to make the human-system conversation more natural. It mainly includes following functions:

— active silence breaking;

— emotion recognition and expression;

— dynamic interruption;

— mood response;

— topic changing;

— conversation delay/disfluency;

— asymmetric dialogue.

Conversation disfluency is a generally recognized phenomenon that happens in the spoken utterance. When implementing FDX speech interaction, conversation disfluency needs to be considered and can be used to mimic natural speech.

EXAMPLE Using FDX speech interaction UI, a system can implement an active utterance response when a user stops talking or chooses to be silent as a listener when the user keeps talking.

6.3.5 Speech synthesis

Speech synthesis refers to conversion of data to speech that represents the content of the data. In an FDX speech interaction UI, the function of conversion of text to speech called TTS is characterized by its adaptive and natural voice output. It can be recognized as a reverse process of ASR and is composed of the following three parts:

a) text analysis, used to extract textual features and transform a grapheme into a phoneme based on a phoneme dictionary;

b) prosody analysis, used to predict the fundamental frequency, duration, tone, intonation, speed and other prosodic features;

c) acoustic analysis, used to implement the mapping from textual parameters to speech parameters, and finally the speech is synthesized by a vocoder.

Common approaches for TTS include waveform splicing and parametric synthesis. The former involves extracting appropriate splicing units from the corpus and splicing them into sentences. The later requires parametric modelling of a phoneme dictionary and uses ML approaches to predict prosodic and acoustic parameters.

6.4 Resources

6.4.1 Knowledge base

Depending on the application, FDX speech interaction UI needs to be operated by relying on human knowledge, at the very least for the conversation processing component. Knowledge can also be acquired during the operation stage. Once acquired, new knowledge combined with an existing knowledge base can be applied to the defined interaction task.

From an information processing psychology perspective, knowledge can be categorized into declarative knowledge and procedural knowledge. Declarative knowledge is knowledge about what is. It is easy to be verbalized and translated into statements and thus it can be regarded as explicit knowledge. Procedural knowledge is knowledge about how to do something. It is often hard to be verbalized and described and thus can be regarded as tacit knowledge. The types of knowledge in this document are not exclusive. There are other taxonomies of knowledge such as episodic knowledge, abstract knowledge, factual knowledge, conceptual knowledge and metacognitive knowledge.

EXAMPLE A proposition of “the capital of China is Beijing.” is declarative knowledge, while “how children learn to speak” is procedural knowledge.

NOTE Within the AI domain, procedural knowledge can be regarded as a programming intelligence that describes how to do something for AI. This kind of programming includes many different programs that the AI system can execute and then allows the system to complete these tasks in a declarative knowledge-based AI system. Instead of specific processes, the system knows what it can do and then assists the program to utilize the appropriate knowledge in an effective way.

Knowledge technologies for FDX speech interaction UI, e.g. ontology, semantic web, knowledge graph and knowledge engineering, are widely applied in domains where expert knowledge is needed. In an FDX speech interaction process, human knowledge is collected and represented as a set of concepts, semantic relationships and axioms. They are coded into machine readable format using, e.g. extensible markup language (XML), resource description framework (RDF) and web ontology language (OWL). Knowledge is often engineered with graphic-based models (e.g. knowledge graph) and databases (e.g. NoSQL database) for specific conversation topics. Such knowledge techniques embedded within the FDX speech interaction UI can help the system to obtain human knowledge, make inferences in a transparent way and, therefore, support a better conversational performance. More information for knowledge engineering and representation are described in ISO/IEC TR 24372[8].

6.4.2 Data resources

Data resources for an FDX speech interaction process can include scenario data, business data, user data and historic data. Such data are important for each functional component described in 6.3 and for training, testing and operating of FDX speech interaction UI. Some FDX speech interaction UIs can use "continuous learning", where the input data and the resulting action are also used to update the database held by the FDX speech interaction UI while the FDX speech interaction UI is operating.

6.5 Computing infrastructures

6.5.1 Cloud and edge computing

FDX speech interaction can use cloud and edge computing to provide better service. For example, ASR and conversation processing can be deployed on the cloud which offers a strong computing capability. For the front-end component, the edge computing can be used to enhance performance. Figure 7 exemplifies the deployment of functional components of an FDX speech interaction UI. For cloud computing, technical details are described in ISO/IEC 17789.[2] For edge computing, more information can be found in ISO/IEC TR 23188.[6]

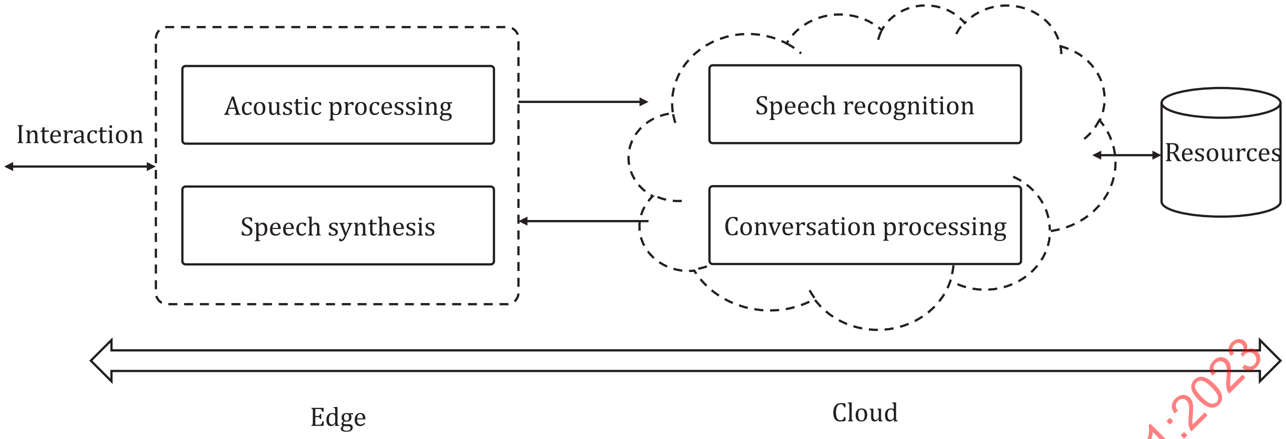


Figure 7 — Example deployment of functional components of FDX speech interaction UI

6.5.2 AI and ML systems

In recent years, AI and ML systems have been applied to support all kinds of intelligent devices and services. The deep neural network based ML systems approach (e.g. RNN, LSTM, CNN) can be used for knowledge acquisition and model training, validation and verification. Transformer-based language models, such as bidirectional encoder representations from transformers (BERT), generative pre-training (GPT) and switch transformers, are typical deep learning techniques used for NLP tasks. The transformer model is characterized by attention, self-attention and positional encodings, which allows a massively parallel computing for NLP. More information on AI and ML systems are described in ISO/IEC 22989,[3] ISO/IEC 23053[4] and ISO/IEC TR 24372[8].

6.5.3 Network

In case of cloud-edge services, network resources are also essential for supporting FDX speech interaction UI. This requires high bandwidth and low latency.

7 Functional requirements and recommendations of FDX speech interaction UI

7.1 General requirements and recommendations

The FDX speech interaction UI shall:

- be able to operate using natural speech language by human users;
- only need to be triggered at the very beginning of the conversation and shall not have to be triggered during the conversation;
- be able to be intervened with meaningful interruption on purpose at any time during the broadcasting or speaking and shall be capable for proceeding dialogue after being interrupted by human users;
- perform conversations based on the context.

The FDX speech interaction UI should:

- be operated using other manners such as gesture and movement;
- have an icon or a cue (in visual or audio) on itself that reminds a user of whether FDX speech interaction is operating;
- be capable of providing visual or audio feedback of the result of the ASR that can be accessible and editable or modifiable by a user;

NOTE 1 If the device is screen-attached, the result of the ASR can be displayed on the screen for visual accessibility.

NOTE 2 This function is useful when the system helps a user for recording the information using speech interaction, thus the user can implement some degree of oversight and control on the result.

— have a physical or a virtual button on itself that controls the interaction functions and sets up the preferences;

NOTE 3 Settings can include the function setting, language setting and information setting.

— adapt to diverse environmental background noises.

7.2 Interaction task requirements and recommendations

All interaction tasks for FDX speech interaction UI shall be well-defined and have a clear domain.

Each task should reflect the needs from users.

7.3 Functional component requirements and recommendations

7.3.1 Acoustic acquisition requirements and recommendations

The acoustic acquisition for FDX speech interaction UI shall:

— use microphone or microphone array in the front-end, to provide functions of continuous audio collection, speech enhancement, acoustic source localization, dereverberation, de-noise, echo cancellation and speech source extraction, as described in 6.3.2;

— be able to implement near-filed audio collection and implement far-field audio collection;

NOTE 1 Near-filed often refers to the distance between microphone and speech source within 1 m, while far-filed refers to a distance that is longer than 1 m and shorter than 5 m.

— be able to localize the target speaker by calculating the planar angle, azimuth angle, pitch angle and distance between microphone and target speaker and to improve the SNR of speech signals.

The acoustic acquisition for FDX speech interaction UI should:

— be able to set the compression level of speech signals and be adapted for compression and decompression under various coding formats and algorithms without changing the content of speech;

NOTE 2 Common coding formats for speech can include EVRC and Rec. ITU-T G.711[9] and Rec. ITU-T G.723.1[10] series developed by ITU-T. The coding formats for audio can include AAC, AC3, MP3, WMA, WAV.

— be able to identify the speaker using voiceprint recognition techniques.

7.3.2 Speech recognition requirements and recommendations

The speech recognition for FDX speech interaction UI shall:

— provide continuous ASR, semantic VAD and irrelevant content rejection, as described in 6.3.3;

— be available for at least one language;

— be able to process a continuous speech stream;

— be able to recognize the command word and key word used for voice trigger. Both command words and key words can be pre-defined and customized by a user;

— be able to continuously monitor and recognize the speech stream after a voice trigger;

- be able to detect the starting point and ending point for multiple speech fragments from a continuous speech stream;
- be able to set the silence waiting duration between two speech fragments and adjust the sensitivity of VAD;
- reject recognition of inappropriate contents based on the semantics of statement and scenarios.

The speech recognition for FDX speech interaction UI should:

- be able to be adapted to multiple languages;
- support hybrid-language speaking;
- support multiple command words for voice trigger;
- provide function of speaker diarization including speaker segmentation and clustering;

NOTE 1 Speaker segmentation refers to finding out the time boundaries of speaker change among multiple speakers and the audio stream is cut into multiple speech segments according to those boundaries. Speaker clustering refers to categorization of one or more speech segments belonging to the same speaker.

- provide function of post-processing of recognition text after ASR.

NOTE 2 Common post-processing can include character normalization, punctuation prediction, text replacement.

7.3.3 Conversation processing requirements and recommendations

The conversation processing for FDX speech interaction UI shall:

- provide NLU, semantics ranking, data searching and dialogue management, as described in 6.3.4;
- be able to understand the user's intention and deliver some degree of forecast about future conversational contents based on the knowledge (see 6.4.1) and data resources (see 6.4.2);
- operate depending on conversational context, which is supported by various knowledge and data that are described in 6.4.

The conversation processing for FDX speech interaction UI should:

- be able to provide function of reasoning including spatial reasoning, temporal reasoning, common sense reasoning, computed policy application or any form of reasoning than can be coded;

NOTE The function of reasoning can enable the system to understand, forecast and make a decision. It is often supported by an expert system, logic programming and knowledge engineering.

- be able to generate and transform the text into an artificial speech. The content of text can include:
  - a simple reply text;
  - a reply text based on a predefined template;
  - a reply text by understanding and responding to the user's intention;
  - a reply text of reasonable guidance or recommendation;
- be able to track conversational status, manage conversation strategy, and either change or continue with conversation topic. Such functions are based on scenarios, user's intentions and conversation states, as described in 6.3.4.5.

7.3.4 Speech synthesis requirements and recommendations

The speech synthesis for FDX speech interaction UI shall:

- provide TTS, as described in 6.3.5;
- be available for at least one language;
- be able to transform text into continuous speech stream as an output;
- be able to adjust the acoustic prosody, speed, tone, intonation of output speech;
- support multi-timbre synthesis including all genders and ages.

The speech synthesis for FDX speech interaction UI should:

- be able to be adapted to multiple languages;
- support hybrid-language speaking;
- be able to simulate the voice characteristics of the target speaker and output the speech with the auditory perception characteristics of the target speaker;
- output a natural speech with acceptable naturalness and intelligibility. The artificial speech quality can be measured using mean opinion score (MOS). The MOS result of speech synthesis in the FDX speech interaction UI should be evaluated above four points. MOS can be quantified as shown in Table 1.

Table 1 — MOS quantified value and corresponding audiometry effect of synthesized speech

MOS	Audiometry effect of synthesized speech
5	Excellent: not aware of any unnatural speech, close to an announcer.
4	Better: can only detect a small amount of unnatural speech.
3	Acceptable: can be perceived as unnatural speech, but it is acceptable.
2	Poor: obviously aware of unnatural speech, not willing to accept.
1	Very bad: unacceptable.

NOTE MOS is a subjective measurement of speech quality.More details and methods for assessment of speech quality referring to MOS can be found in Rec. ITU-T P.800.1[11] and Rec. ITU-T P.800.2[12].

7.4 Resource requirements and recommendations

Knowledge and data shall be used for context-based conversations. They should be acquired from various sources and be well-stored in the local server or on the cloud.

7.5 Computing infrastructures requirements and recommendations

Computing infrastructures shall support the proper functions and operations of FDX speech interaction UI. Available techniques and infrastructures, such as cloud and edge computing, AI and ML systems, and future network, should be considered and applied. The rate of utilization of infrastructures should remain stable throughout the operation.

EXAMPLE In order to deliver a good performance, many ML approaches have been used for some components of FDX speech interaction UI. LSTM-RNN and deep convolutional neural network (DCNN) can be used for the acoustic modelling. Encoder-decoder approaches can be used for the language modelling. CNN can be used for the beamforming. Bi-directional encoder representations from transformers (BERT) can be used for NLU.

8 Processes of FDX speech interaction UI

8.1 General

This document describes two FDX speech interaction UI processes, the engineering process from inception through retirement, and the interaction process between humans and the system.

8.2 Engineering process

The engineering process provided in this document aims to help stakeholders, including designers, implementers and verifiers, build FDX speech interaction UI more effectively and efficiency. This document does not prescribe a specific pipeline, but provides a general overall process including main engineering stages. Figure 8 provides an example of the stages and high-level process that can be applied to the development.

NOTE Designer refers to the entity that receives data and a problem specification, and creates an AI model. Implementer refers to the entity that receives an AI model and specifies what computation to execute. Verifier refers to the entity that verifies that a computation is being executed and model is performing as designed.

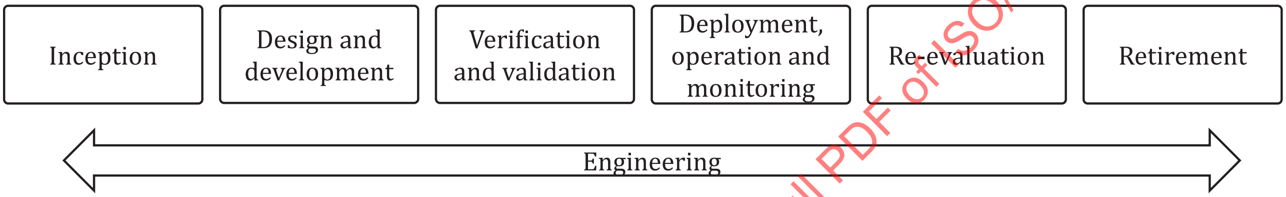


Figure 8 — Example of the engineering process of FDX speech interaction UI

FDX speech interaction UIs can differ from each other depending on the task, scenarios, attaching devices and embedding approaches, which can impact the process stages. For example, most models and functions in FDX speech interaction UI can be trained using ML algorithms but require multiple iterations of improvement to achieve acceptable levels of accuracy and reliability. Therefore, in contrast with conventional rule-based approaches that are programmed to be comprehensible according to requirements and specifications, the testing and verification of FDX speech interaction UI embedded with ML approaches, especially deep learning, can be challenging. A common process includes the following stages.

a) Inception: occurs when stakeholders decide to turn an idea into a tangible system. During the inception stage, stakeholders should determine why an FDX speech interaction UI needs to be developed. What problem does it solve? What scenarios does it fit? What customer needs or business opportunities does it address? The above questions may get answers through a market survey and analysis, and multi-stakeholders with diverse expertise can help to identify the requirements and the costs.

b) Design and development: works at this stage to create the FDX speech interaction UI and concludes with a software or hardware (also APP, SDK, SaaS) that is ready for deployment. During this stage, and particularly before the conclusion, stakeholders should ensure the FDX speech interaction UI fulfils the original objectives, requirements and other targets identified during the inception stage.

c) Verification and validation: works in this stage to check that the FDX speech interaction UI from the design and development stage works according to requirements and meets objectives.

d) Deployment, operation and monitoring: works at this stage to install and configure the FDX speech interaction UI in a target environment. It should be available for use. Normal operation and the running errors and failures may be monitored and reported to stakeholders for action.

e) Re-evaluation: during this stage, the results from operation monitoring should be evaluated against the objectives and the requirements determined for the FDX speech interaction UI. Once the problems are identified, refinement of objectives and requirements should be conducted.