Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 08, No. 04, December 2022.

TOWARD DEVELOPING AN INTELLIGENT PERSONAL ASSISTANT FOR TUNISIAN ARABIC

Inès Zribi[1] and Lamia Hadrich Belguith[2]

(Received: 6-Jun.-2022, Revised: 9-Aug.-2022, Accepted: 27-Aug.-2022)

ABSTRACT

Intelligent systems powered by artificial intelligence techniques have been massively proposed to help humans in performing various tasks. The intelligent personal assistant (IPA) is one of these smart systems. In this paper, we present an attempt to create an IPA that interacts with users via Tunisian Arabic (TA) (the colloquial form used in Tunisia). We propose and explore a simple-to-implement method for building the principal components of a TA IPA. We apply deep-learning techniques: CNN [1]*, RNN encoder-decoder* [2] *and end-to-end approaches for creating IPA speech components (speech recognition and speech synthesis). In addition, we explore the availability and free-dialog platform for understanding and generating the suitable response in TA for a request. For this proposal, we create and use TA transcripts for generating the corresponding models. Evaluation results are acceptable for the first attempt.*

KEYWORDS

Intelligent personal assistant, Tunisian Arabic, Speech recognition, Natural-language understanding, Dialog management, Response generation, Speech synthesis.

1. INTRODUCTION

Technological progress has made a large number of advanced application technologies. Among them, we cite smart systems, including spoken-dialog systems and especially the Intelligent Personal Assistant (IPA). As illustrated by [3], IPA is a speech-compatible software that can be found on a specialized device (e.g. Amazon Echo, Google Dot), a mobile device or a computer. It assists the user by answering questions in natural language, giving suggestions, DOIng tasks, etc. Nowadays, IPAs are becoming essential in human lives and have a powerful effect on our everyday lives. They are able to replace humans in some ordinary cases that are repetitive in nature and can be easily automated, including providing flight information, sport results, weather forecasts, share prices, booking hotels, renting cars, etc. [4]. IPAs are designed to accept spoken dialog, which is a natural mode of communication, or typed input in a natural language [5]. Some of them give responses to queries by voice and/or text messages. The architecture of most of them is based principally on five principal modules [6]: speech-recognition module (SR), natural-language understanding module (NLU), dialog-management module (DM), natural-language generation module (NLG) and finally, speech-synthesis module (SS). The quality of an assistant is based principally on the quality of each component. Figure 1 presents the basic architecture of voice IPA inspired from [6].

Apple's Siri, Amazon's Alexa, Google Assistant and Cortana from Microsoft are the most popular and used IPAs developed to help users do some usual and simple-to-complex tasks. They are now a signature feature of some smartphones and tablets. There is also a set of free and open-source assistants, such as Mycroft Core[1], Open Jarvis[2], etc. With the development of deep-learning techniques, many researchers have developed specialized IPAs. Some of them have an object to build a social relation with the user [7]. The IPA determines user's goals and preferences, so that it can recommend conferences to attend and people to meet. Some other IPAs have a goal to check a patient's health indicator [8], support human operators to empower operators in industry environments etc. [9]-[10]. Despite the continuous development of this type of technology, IPAs cover a limited set of languages. They differ from one another. Due to the variety and differences of language dialects, each dialect needs a distinct linguistic model [11]. Therefore, only some dialectal forms are also considered by some IPAs. English, French and Chinese are the most commonly treated languages by the majority of IPAs. However, their

[1] https://mycroft-ai.gitbook.io/docs/mycroft-technologies/mycroft-core
[2] https://openjarvis.com/

1. I. Zribi is with MIRACL Laboratory, Sfax University, Sfax, Tunisia. Email: ineszribi@gmail.com
2. L. H. Belguith is with Faculty of Economics and Manag. of Sfax, Sfax Uni., Tunisia. Email: lamia.belguith@fsegs.usf.tn

performances deteriorate when Arabic is the used language. Table 1 presents the languages treated by the four IPAs: Cortana, Siri, Alexa and Google IPAs. We remark that Modern Standard Arabic (MSA) is considered by some IPAs, while the colloquial form is neglected. Dialectal Arabic (DA) is mainly spoken and used in daily communication. It is used in chat, utilities, radio, phone conversations and so on. As a rule, Arabs are unable to speak the standard form of their language on a day-to-day basis. Therefore, they interact with IPAs using a foreign language (e.g. English for Anglophone persons or French for Francophone persons). So, it is important to develop a spoken-dialog system able to understand the DA.

In this paper, we investigate the possibility to build an Intelligent Personal Assistant for dialectal Arabic, especially, Tunisian Arabic (TA). We explore a simple-to-implement method for building TA IPA components with the availability and free resources (corpus, GPU, APIs, etc.). We apply deep-learning techniques: CNN [1], RNN encoder-decoder [2] and end-to-end approaches for creating IPA speech components (SR and SS). In addition, we use the available and free-dialog platform for understanding and generating the suitable response in TA for a request. For this proposal, we build about 5 hours of TA speech corpora composed of IPA requests. To the best of our knowledge, our work is the first attempt to build IPA-system components for TA. Indeed, no work has been done to building speech synthesis and language understanding and generation for TA. Furthermore, only TA transcripts are used for generating the different models.

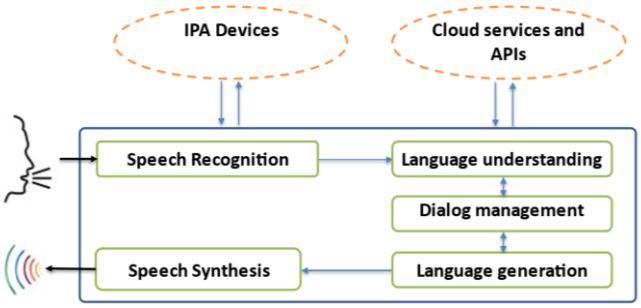

Figure 1. Basic architecture of voice IPA.

This paper has five main sections. Section 2 describes our motivations and some challenges in building an IPA for TA. Section 3 presents an overview of previous work that studied the building dialog systems and the speech components of an IPA. In Section 4, we present our proposed method and the evaluation results. Finally, in Section 5, we present the conclusion and expose our future directions.

2. BACKGROUND

2.1 Motivations for Developing IPA for Tunisian Arabic

2.1.1 Tunisian Arabic

In this paper, we intend to develop components for an IPA that understands and speaks Tunisian Arabic (TA); a dialect of the North African (i.e., the Maghrib) dialects spoken in Tunisia by approximately twelve million people [12]. Although TA is mainly spoken, it is written in social networks, blogs, some novels, as well as in comics, commercials, some newspapers and popular songs. It was influenced by other languages, such as Berber, French, MSA, Turkish, Italian, Maltese, etc. [12]. This is a result of the position of Tunisia between the two continents (Africa and Europe), as well as the variety of civilizations that ruled it and its openness to neighboring cultures. However, code-switching (between MSA, French and TA) is the main characteristic of the TA [13]. For example, the sentence presented in Table 2 is composed of two French phrases "C'est vrai" and "donc", a TA phrase " اللي المدير أســـتاذ أما ما ينجمش يبراتجى ويوصـــل المعلومة" and an MSA phrase "لا يصـــلح لتســـيير الشـــركة". This phenomenon allows the introduction of new words (nouns and verbs) derived from foreign languages (e.g. يبراتجى (ybrAtjý[3]) '*He shares*' derived from the French verb "partager"). Indeed, there are many differences as well as similarity points between TA, Arabic dialects and MSA at different levels: lexical, morphological,

[3] Transliterations of Arabic words are presented in the HSB scheme [83] and are presented between (...). Phonological transcriptions are presented between slashes /…/.

syntactic and phonetic (for more details, see [12]). In addition, TA is distinguished by the presence of words from several other languages. The standard form of the Arabic language is used for some IPAs. While the MSA is the official language of all Arab countries, Arabs do not use the standard form in their daily communications. Colloquial Arabic or dialectal Arabic (DA) is the mother tongue spoken daily by everyone [12].

Table 1. Languages and dialects recognized by devices of four IPAs: Cortana, Siri, Alexa and Google.

Languages	**Cortana**	**Siri**	**Alexa**	**Google**
English	S + D[4]	S + D	S + D	S + D
Portuguese	S	S	S	
French	S + D	S + D	S + D	S + D
German	S	S + D	S	S + D
Italian	S	S + D	S	S
Spanish	S + D	S + D	S + D	S + D
Chinese	S + D	S + D		
Japanese	S + D	S	S	S
Arabic		S		
Turkish		S		
Thai		S		
Swedish		S		S
Russian		S		
Norwegian		S		S
Hebrew		S		
Korean		S		S
Malay		S		
Cantonese		S		
Danish		S		S
Dutch		S + D		S
Finnish		S		
Hindi			S	S

Table 2. Example of TA sentence.

TA	C’est vrai اللي المدير أستاذ أما ما ينجمش يبراتجى ويوصل المعلومة donc لا يصلح لتسيير الشركة
Transliteration	C’est vrai Ally Almdyr ÂstAð ÂmA mA ynjmš ybrAtjý wywSl Almçlwmħ donc lA ySlH ltsyyr Alšrkħ
Translation	*It’s true that the director is a professor, but he can't share information; so, he is unfit to manage the company.*

2.1.2 Importance of Using Dialectal Arabic for an IPA

As we said before, an IPA is a software solution designed to help people in their everyday lives to do multiple tasks, going from simple (e.g. checking mail, making calls, etc.) to complex tasks related to smart home (e.g. opening TV, etc.). To be useful to users, the communication mode should be simple. The Arabic language that is currently supported by some IPAs (e.g. Siri on Apple) represents the standard form of Arabic, while some Arabs (not to say the majority of Arab people) are not fluent in MSA for many reasons. Therefore, they even use an IPA in foreign languages (e.g., French for French speakers, English for English speakers, etc.). Indeed, the absence of the colloquial form of Arabic in all APIs makes the use of such types of technologies relative only to intellectualized people. Therefore, the design and development of an IPA speaking dialectal Arabic will help Arabs interact easily and encourage them to use an intelligent assistant.

2.2 Challenges in Building Tunisian Arabic IPA

2.2.1 Rarity of Resources

The presence of spoken and annotated corpora is important for building an IPA. The automatic processing of TA is a new area of research. Unlike MSA, TA suffers from the scarcity or even the

[4] ‘S’ means the standard form of the language and ‘D’ means the dialectal form of the language.

absence of freely available corpora. The few TA resources developed over the last few years are still in the early stages. Their size is relatively limited compared to that of MSA. The only TA spoken corpus accessible is that of [14].

2.2.2 Ambiguity

Like MSA, TA is characterized by ambiguity at many levels: phonological, morphological, semantic and syntactic. A word or an expression can be understood differently based on the context. For example, the greeting expression السلام عليكم (AlslAm ٍlykm) is also used for '*the goodbye*'. Also, the word باهي (bAhy) can mean '*ok'* or '*good'*. Ambiguity affects the performance of TA IPA because of the confusion in understanding some questions and/or answers. Some cases of ambiguity can be easily addressed, while others require complex disambiguation methods to improve TA IPA achievement. A few works [15] considered the task of disambiguation by TA. This issue complicates the task of building an IPA for TA.

2.2.3 Sub-dialectal Variation

TA is characterized by the existence of many sub-dialectal varieties [15]. The sub-dialects differ at many levels. The same word is pronounced in different ways (e.g. بقرة (baqraħ) '*cow'* is pronounced as /bagra/ and /baqra/). The phonological differences complicate the development of speech recognition and speech synthesis for TA. Similarly, the sense of a word differs from one sub-dialect to another. For example, the word ربح (rbH) means in some sub-dialects '*salt'* and in others '*benefit'*.

2.2.4 Code Switching

Code switching between TA and other languages causes many problems for the development of TA IPA. First, the SR component is not able to distinguish words in TA from other languages. As a result, it transcribes all words using the same script. This creates ambiguities for the NLU module, because a word in French transcribed in Arabic letters can have a different meaning. For example, the French word "merci" '*thank you*' transcribed in Arabic letters can refer to a person's name مرسي (mrsy) '*Morsi*'. Tunisians also use some French words in everyday communication without any modification (e.g. "mécanicien"). This can cause, also, some problems for the SS module.

3. RELATED WORKS

3.1 Dialog System

In general, dialog systems (IPA, chatbot[5], etc.) can be classified into task -oriented systems and task-non-oriented systems [16]. Task-oriented systems (e.g. IPA) try to help the user achieve certain tasks. Task-non-oriented systems (e.g. chatbot) talk to the user to provide responses and entertainment. While developing a dialog system, four methods are proposed for understanding and generating language according to its goal. For the first type of dialog system; oriented task, Chen et al. (2018) [16] have classified methods into two categories: pipeline methods and end-to-end methods.

The typical structure of a pipeline methods consists of four key components:

- The language-understanding component (NLU) parses the user utterance into pre-defined semantic slots. It classifies the user's intent and the utterance category into one of the pre-defined intents. The NLU component extracts important information, such as named entities and fills the slots. Deep-learning techniques are successfully applied in intent classification. Hashemi et al. [17] have applied Conventional Neuronal Network (CNN) in intent classification, while Sreelakshmi et al. [18] have used Bi-Directional Long Short-Term Memory (Bi-LSTM) networks for intent identification. Slot filling and named-entity extraction are important tasks for NLU components. Deep-belief networks (DBNs) are usually used by some researchers, like [19]. CNN has also been exploited in slot filling by [20]. Pre-trained BERT and BiLSTM have been employed by [21] for intent and argument detection.

[5] A chatbot is a program that allows a human–computer conversation to be conducted *via* auditory or textual methods using natural language[40], [84]. It operates almost as an IPA.

- The dialog state tracker is the main component in a dialog system. It divines the objective of each turn of dialog. For tracking dialog state, [22] exploited rule-based methods. [23] and [24] made use of statistical and deep-learning techniques.
- Dialog policy learning learns the next action based on the current dialog state. Like in previous components, rule-based [25], statistical and deep-learning approaches [26] have been applied.
- Natural Language Generation (NLG) is responsible for generating the response. As illustrated in [16], conventional approaches are widely used in NLG. It transforms the input (i.e., semantic symbols) into an intermediary structure (such as tree-like or template structures) and then, the intermediate structure is transformed into the final response [27]. Deep-learning techniques, such as LSTM-based structure, are proposed by [28] and [29] to NLG. Wen et al. [28] used a forward RNN generator together with a CNN re-ranker and a backward RNN re-ranker [16]. Zhou et al. [30] adopted an encoder-decoder LSTM-based structure to generate correct answers based on the question information, semantic slot values and dialog act type. The sequence-to-sequence approach is used by [31]. It can be trained to produce natural-language strings as well as deep syntax-dependency trees from input dialog acts. Recurrent neural-network language generation (RNNLG) is proposed by [32]. It can learn to generate statements directly from dialog-act pairs of statements with no pre-defined syntax and no semantic alignment.

End-to-end approaches to develop dialog task-oriented systems have been proposed and used by several researchers [33]–[35]. They have combined several methods, like an encoder-decoder model, an end-to-end reinforcement learning technique, an attention-based key-value retrieval mechanism, etc. All of the end-to-end methods use a single module and interact with structured external databases. The input of the model is the user request and the output is the response.

In each presented approach, there are multiple used techniques: parsing, pattern matching, Artificial Intelligence Markup Language (AIML), chatscript, ontologies, Artificial Neural Network Models, etc. Several commercial, free platforms, APIs and libraries have used these techniques for understanding natural language, dialog management and language generation in order to develop conversational systems. Among these platforms, we cite Dialogflow [6] from Google, IBM Watson Assistant [7], Pandorabots[8], Rasa [36], etc. Some of these platforms are exploited in developing some Arabic chatbots for MSA [37], [38] and Colloquial Arabic (BOTTA [39], Nabiha [40], etc.). The majority of previous efforts in creating Arabic dialog systems have been focusing on developing task-non-oriented dialog systems. In contrast, in this work, we focus on building a task-oriented dialog system using Rasa platform, which is able to understand TA and DOIng some simple tasks. Furthermore, to the best of our knowledge, there is no work dealing with creating a TA task-oriented dialog system or developing Tunisian IPA, where our work is the first one.

3.2 Speech Components

We present in this sub-section some work proposed for Speech Recognition (SR) and Speech Synthesis (SS) for Latin and Arabic languages. As defined by [41], SR is an automatic way to transcribe speech into text. It is used to make machines understand human speech. SS has the inverse task. It transforms text into voice. In general, the SR module receives an IPA user request and the SS gives the response to the user.

3.2.1 Speech Recognition

Like the dialog system, the automatic speech recognition (ASR) can be classified into two approaches: conventional ASR pipeline approach and end-to-end ASR approach. The conventional ASR pipeline includes trained acoustic, pronunciation and language model components which are trained independently. It regroups classical methods: *(1)* rule-based methods that use phonetic rules in order to convert graphemes into phonemes, *(2)* probabilistic and data-driven methods ([42], [43], etc.), which are based on a phonetic dictionary, acoustic models and feature-extraction step. These methods utilize Hidden Markov Models (HMMs), Conditional Random Fields (CRFs), Deep-learning techniques, etc. to generate acoustic models and extract features. End-to-end voice recognition goes a long way to

[6] https://dialogflow.cloud.google.com/
[7] https://developer.ibm.com/articles/introduction-watson-assistant/
[8] https://www.pandorabots.com/

simplif the complexity of traditional speech recognition. It is based on deep-learning techniques. No preliminary steps (phonetic rule construction, acoustic dictionary, feature extraction, etc.) are required. In this approach, we need a set of records and their transcriptions and the deep neural network can automatically learn language or pronunciation information. Among the works conducted using this approach, we cite [44]–[49].

Few works are conducted in Arabic due to the scarcity of transcribed speech corpora. [50], [51], etc. have proposed and tested SR classical methods which are based on HMMs. The broadcast news-transcription system proposed by [50] has two main components: an audio partitioner and a word recognizer. Data partitioning is based on an audio stream-mixture model and divides the continuous stream of acoustic data into homogeneous segments [50]. The second component of the proposed system determines the sequence of words for each speech segment. The recognizer utilizes continuous-density HMMs for acoustic modeling and the n-gram statistics for language modeling. Lamel and Gauvain [50] have developed a pronunciation lexicon based on a grapheme-to-phoneme tool. It contains 57k distinct lexical forms from 50 hours of manually transcribed vocalized data. The language model contains up to 4-gram. The same method has been adopted by [51]. Their proposed system is based on Carnegie Mellon University's Sphinx tools. It uses 3-emmiting state HMMs for triphone-based acoustic models. The system was trained on 4.3 hours of Arabic broadcast news corpus and tested on 1.1 hour. The phonetic dictionary contains 23,841 definitions, corresponding to about 14,232 words. The language model contains both bi-grams and tri-grams. The same approach has been adopted by [42] for recognizing TA speech. The author has developed a phonetic dictionary for TA based on the CRF algorithm. Then, the dictionary is integrated into the MSA SR system. Ben Ltaief et al. [52] have described an SR system for TA based on the Kaldi toolkit. They have built 2 acoustic models: HMM-GMM (Gaussian mixture models) and HMM-DNN and have trained 3-gram models. Their models are based on the TARIC corpus [53]. Messaoudi et al. [54] have exploited the DeepSpeech architecture [2] to generate a model to recognize TA speech. They have exploited four Arabic corpora (two TA corpora and two MSA corpora) for generating language and deep-learning models. We note that these SR models are not available for testing and using.

The majority of previous efforts in creating Arabic ASR have been focusing on using conventional traditional pipeline. The only work done for TA based on end-to-end approach is proposed by [54]. In this work, we propose to follow the same approach. The main difference between our work and that of [54] is the nature of our corpus, which is based only on TA corpus.

3.2.2 Speech Synthesis

There are principally two types of speech-synthesis approaches. Classical approaches are based on Concatenate Speech Synthesis (CSS) and HMM techniques. The principle of CSS is to generate speech by concatenating its units one after the other [55]. The generation requires a corpus composed of utterances with well-annotated phones. The quality of generated speech is related to the quality of the collected corpus. The HMM synthesis techniques, called statistical parametric synthesis of speech [55], extract parameters from the recorded utterances which are then used to generate speech. The quality of the generated voice is maintained, even if the size of the training corpus is small. The classical approaches are used in SS for several languages, such as English ([56], [57]), French ([58], [59]), etc. and MSA ([60]–[63], etc.).

The second approach is based on deep neural architectures that have proved successful at learning the fundamental features of data [64]. Several architectures are proposed. Among the most famous ones, we cite WaveNet [65]; a deep generative model of audio data that operates at the waveform level. The application of this model to SS shows that produced samples surpass many SS systems in subjective naturalness. However, it has some drawbacks. First, the model is not a full end-to-end system. Second, the generation of speech is very slow [64]. Deep voice [66] is another deep neural architecture used for SS. It is an end-to-end neural architecture. Traditional text-to-speech pipelines inspire it, but its components are replaced with neural networks. It is simpler than classical approaches. Any human involvement is required for deep voice model training. Tacotron [67] is another end-to-end architecture for SS. It is a generative model based on a seq.-to-seq. model with an attention mechanism [68] that produces audio waveforms directly from the characters. Tacotron automated some SS tasks, such as

feature engineering and human annotation. Tacotron 2 is an improved version of Tacotron proposed by [69]. It eliminates non-neural network components used to synthesize speech, such as the Griffen-Lim reconstruction algorithm [64]. Shen et al. [69] have used hybrid attention [70] with a recurrent seq.-to-seq. generative model and a modified wavenet acting as a vocoder to synthesize speech signals [64].

The deep approach was proposed and tested for several languages, such as English. In the last few years, a few researchers have tested some architectures for MSA. Tacotron 2 [69] has been tested for a voweled MSA corpus by [64]. Hadj Ali et al. [71] have tested DNN for the task of grapheme-to-phoneme conversion using diacritized texts. Abdelali et al., [72] have also tested Tacotron [67], Tacotron 2 [69] and Model ESPnet Transformer TTS [73] in the Arabic language. To the best of our knowledge, there is no work being done for TA and our work is the first one developing an SS for TA. In Table 3, we present a comparison between speech works (SR and SS) done for Arabic language.

Table 3. Comparison between different Arabic speech systems.

	Ref. No.	Approach	Classification method	Used dataset	Result	MSA/TA
Speech Recognition	[50]	Conventional pipeline	HMM + n-gram	1200 hours of broadcast news data	WER = 0.209	MSA
	[42]	Conventional pipeline	GMM-HMM model + n-gram	10 hours of TA	WER = 0.226	TA
	[51]	Conventional pipeline	HMM	4.5 hours of Arabic TV news	WER = 0.09	MSA
	[52]	Conventional pipeline	HMM- DNN + HMM-GMM	10 hours of TA	WER= 0.368	TA
	[54]	End-to-end	RNN encoder-decoder	61 hours and 34 minutes of MSA and TA	WER = 0.244	MSA + TA
Speech Synthesis	[63]	Classical approach	HMM	598 utterances	MOS = 4.86	MSA
	[64]	Deep approach	Sequence-to-sequence architecture + flow-based implementation of WaveGlow	2.41 hours	MOS = 4.21	MSA
	[71]	Deep approach	DNN	1597 utterances	MAE[9] = 19	MSA
	[72]	Deep approach	Model ESPnet Transformer	9969 utterances male and female voices	MOS = 4.40	MSA

4. TUNISIAN IPA COMPONENTS

An IPA usually operates by the following these steps. First, when it is on and is not used for a certain time, it goes into a "*listening mode"*. When the user calls the IPA by pronouncing the Trigger Word (TW) (e.g. "*Alexa*", "*Siri*", "*Hey Google*", etc.), the latter wakes up. It waits for the user's request. Then, the IPA accomplishes the requested task and gives vocal response to the user. Finally, it goes back into the listening mode. Hence, we propose to build two SR modules. The first one (SR-TW), based on a Convolutional Neural Network [1], is responsible for detecting the trigger word (TW). When it is recognized, the second module (SR-R), based on the DeepSpeech architecture [2], is activated for receiving and transcribing users' requests. Once the request is received, the Language Understanding module is activated. It is responsible for classifying the intents and detecting the entities. The latter are used by the Dialog Management model to decide the next action to do. Then, the Language Generation module prepares and generates the suitable response. For these three components, we propose to apply the RASA dialog framework to generate the response by following the dialog history of the user and generate the response according to user intention. It also accomplishes the requested task. Finally, the generated response will be sent to the Speech Synthesis (SS) model in order to generate the corresponding voice. We apply the Tacotron 2 [69] model to the TA. Figure 2 presents the architecture of our IPA. We note that these components are dedicated to recognizing and understanding the commands of the users relative to four basic IPA skills: greeting and knowledge, weather forecasts, checking email and asking time and date. We present, in the rest of this section, the details of our proposed method.

[9] Mean absolute error is a measure of errors between paired words expressing the same speech.

4.1 Speech Recognition

4.1.1 Proposed Method

We present in this sub-section our proposed methods for two SR modules. The first one, Speech Recognition-Trigger Word (SR-TW), is responsible for detecting the trigger word (TW). The second module, Speech Recognition-Request (SR-R), is responsible for transcribing users' requests.

Speech Recognition-Trigger Word (SR-TW): In order to activate the IPA, a TW should be said by the user. We propose to generate a model that classifies short sounds (1 second) into two classes: TW and non-TW. For classification, we apply the deep neural network, in particular the Convolutional Neural Network (CNN). Its architecture is composed of eight hidden layers: an input layer, four convolutive layers followed each one by a pooling and drop layer, one flatten layer, two dense layers followed each one by the dropout layer and finally, an output layer. This architecture is often used for recognizing and classifying speech. "Hey Cortana", "Hey Google" and "Alexa" are some of the TWs, respectively, used by Microsoft Cortona, Google and Alexa IPAs. We have chosen عالسلامة (çAlslAmħ) '*hello'* as a TW.

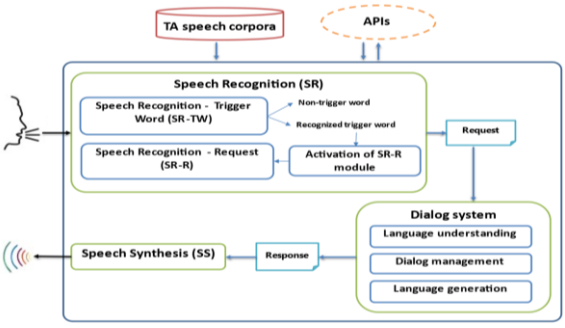

Figure 2. Architecture of our IPA.

Speech Recognition-Request (SR-R): For recognizing user requests, we applied the deep-learning architecture proposed by [2], baptized DeepSpeech. We chose to apply this architecture, because it has shown its efficacy for several languages (English [2], Mandarin [74], German [47], etc.). This architecture also has shown its efficacy for TA [54]. We note that the SR module proposed by [54] is not available for testing and using.

DeepSpeech is also robust when it is applied in noisy environments [2]. With deep learning, we do not need to do the extraction of features or generate the phonetic dictionary. The DeepSpeech architecture is composed of five hidden layers [2]. The three first layers of non-recurrent $h_t^{(1-3)}$ (dense) are fully connected and computed by the ReLu activation function. The fourth layer is a bi-directional recurrent layer. It includes two sets of hidden units: a set with the forward recurrence $h_t^{(f)}$ and a set with the backward recurrence $h_t^{(b)}$. The fifth (non-recurrent) $h_t^{(5)}$ layer takes both the forward and backward units as inputs. The output layer is a standard softmax function that yields the predicted character probabilities for each time slice *t* and character *k* in the alphabet. Further, the Connectionist Temporal Classification (CTC) loss function is used to maximize the probability of correct transcription [75].

We do not modify the architecture of DeepSpeech, because it has shown its performance for complex languages with a large number of characters, like Mandarin [45]. For training an automatic SR system based on a DeepSpeech system [2], two main components are used: a Recurrent Neural Network (RNN) and a language model. We used a set of audio files with their corresponding transcriptions in order to train the RNN model. For the language model, we have used KenLM to generate an n-gram model [76]. We have used the same values of parameters as proposed in [54]. We have generated a 3-gram model with an alpha value of 1 and a beta value of 1.5. We have used an alphabet composed of 38 characters and omitted the short vowels of the alphabet.

4.1.2 Dataset

We exploited in the development of IPA components the freely available spoken corpora for TA. To the best of our knowledge, the Spoken Tunisian Arabic Corpus (STAC) [14] is the only publicly available

spoken corpus. It is composed of 5 transcribed hours collected from different Tunisian TV channels and radio stations. It contains spontaneous speech, less spontaneous speech and prepared speech and a large number of speakers (about 70 speakers) in order to make the dataset a representative sample of the TA. We have exploited a part of STAC (2 hours and 31 minutes). Some pre-processing steps are done on this corpus. First, we have removed all types of annotations, such as disfluencies, etc. We have, then, corrected some orthographic errors. We have corrected them according to the CODA convention [12]. Next, we have removed unclear speeches, music, superposed sounds and long pauses. After that, we have subdivided the audio files into small audio waves (less than or equal to 10 seconds). Finally, we have converted files into wave format with a mono audio channel and a sample rate of 16,000 Hz in order to be read by the DeepSpeech pipeline. We obtained, after pre-processing, 1 hour and 56 minutes of pure transcription.

We recall that our objective is to build an IPA for TA. So, we have enriched the corpus with some transcriptions of an IPA user command, such as greeting and acknowledgement, providing weather forecasts, asking for the date and time and finally, checking new email. We have recorded commands for 28 persons (3 men and 25 women). We have augmented the transcriptions by adding noise and modifying the pitch. We have obtained a total of 50 minutes. We have also enriched the corpus with some other transcriptions (Tunisian dialect stories and some read chapters from the Tunisian constitution in TA). The total size of these transcriptions is 1 hour and 27 minutes. We have augmented this corpus by adding noise and modifying the pitch (down and up) of its files. Table 4 summarizes the details of our corpus.

Table 4. Size of our corpus used in SR modules.

Corpus	Size
A part of STAC	1 hour and 56 minutes
IPA corpus	50 minutes
Other transcriptions	1 hour and 27 minutes
Augmentation	3 hours and 24 minutes
Total	**7 hours and 37 minutes**

For training and testing SR-TW, we have collected, from the corpus, transcriptions that contain the word عالسلامة (çAlslAmħ) '*hello*'. The duration of each transcription is about 1 second. We have collected multiple pronunciations (10 persons) of the trigger word. We have augmented the corpus by adding noise and modifying the pitch. We obtained about 16 minutes of different pronunciations of the trigger word. For the class non-trigger word (NTW), we have collected different sounds that have a duration equal to 1 second, pronouncing different words in TA. We have obtained about 33 minutes. To train and test the TW-SR model, we have divided the corpus into 70-30%. In contrast, we used the division 80-10-10% for the training, validation and testing of the SR-R models. For generating n-gram models, we have exploited the TA corpora used in [77], composed of 260,364 words.

4.1.3 Evaluation

To measure the performance of our module, we have calculated the Word Error Rate (WER) and the Character Error Rate (CER) for SR-R. A Lower WER respectively CER is often used to indicate that the Speech Recognition model is more precise in recognizing speech. A higher WER respectively CER, then, often associated with a lower accuracy. Since the SR-TW classifies short sounds into TW and Non-TW, we have calculated the accuracy measure to test the accomplishment of this component. Formulae of the following measures are presented below, where Nw is the number of words in reference text, Sw is the number of words substituted (a word in the reference text is transcribed differently), Dw is the number of words deleted (a word is completely missing) and Iw is the number of words inserted. We note that the formula for CER is the same as that of WER, but CER operates at the character level instead. Table 5 presents the results of the two models. The accuracy value of SR-TW is an encouraging result. The errors are related to some homophones, such as عالسلامة (çAlslAmħ) '*hello' et* بالسلامة (bAlslAmħ) '*bye'*. For SR-R, the evaluation results of [54] are better than our results. This is due to the size of their used corpus that contains STAC corpus and other speech TA corpora. By analyzing the results, transcription errors are caused by the insertion of some extra letters. The presence of homophones, disfluencies, etc. are the principal causes of failure cases.

(1) $Accuracy = Number\ of\ correct\ predictions\ /Total\ number\ of\ predictions$

(2) $WER = \frac{(Sw + Dw + Iw)}{Nw}$ (3) $CER = \frac{(Sc + Dc + Ic)}{Nc}$

Table 5. Evaluation results of the two SR models.

Model	WER	CER	Accuracy
SR-TW	-	-	0.97
SR-R	0.41	0.30	-
Tunisian DeepSpeech [54]	0.322	0.204	-

4.2 Natural Language Understanding, Dialog Management and Response Generation

4.2.1 Proposed Method

Over the last decade, there has been a focus on using statistical and machine-learning methods in language understanding, dialogue management and language generation rather than traditional technologies (i.e., rule-based method). Indeed, we propose to apply a statistical method to understand requests, manage dialogs, generate a suitable response and do the task. Therefore, we have used the Rasa framework [36]: an open-source framework which allows developers to create a machine learning-based conversational system (especially a chatbot). Rasa proposes two main modules: Rasa NLU and Rasa Core. Rasa NLU analyzes the user's request. It classifies it based on the appropriate intent and then extracts the entities. Rasa Core chooses the action that the dialog system should take based on the output of the Rasa NLU (structured data in the form of intents and entities) using a probabilistic model. Rasa leads to creating and generating models DOIng simple and complicated tasks in an efficient way, even with minimal initial training data [36]. It regroups a set of components that make up the NLU pipeline (tokenization, entity extraction, intent classification, response selection, pre-processing and more) and works in succession to process the user input into a structured output. It also has a set of policies that manage conversation actions. Both policies and components are based on machine learning (e.g. SVM, CRF, RNN, LSTM, etc.) and rule-based techniques. We have chosen to use Rasa for many reasons. First, it is free and an open-source tool. It can run locally [78]. Second, the use, implementation and bootstrapping of Rasa are relatively easy [79]. Since Rasa NLU does not support the Arabic language, we have applied the pre-configured NLU pipeline. It is composed of eight components. Rasa Core uses policies to decide the next action in a dialog conversation. It provides rule-based and machine-learning policies. In our work, we have also used pre-configured policies. Figure 2 presents the components of the pre-configured pipeline and pre-configured policies. The full description of the components and policies is presented in the documentation of Rasa [80].

Our training data is composed of a list of messages that IPA expects to receive. These messages are annotated with intents and entities that the RASA NLU learns to extract. As we said before, our IPA is limited to four services: "greeting and knowledge", "weather forecasts", "checking email" and "asking time and date". Therefore, our corpus includes intents for these services. We added other basic intents; namely, "affirm", "goodbye", "thanks", "person identification" and "city identification" to ensure a good conversation. Our training data also contains a set of responses that the user expects to receive. We have defined five types of responses: "bye", "end", "start", "first conversation" and "thanks". We have added four customizable responses to the intents: "ask mail", "provide weather forecasts", "person identification" and "ask date and time". We identified and annotated nine entities; namely, "date", "component of the date", "mail", "person's name", "time", "Tunisian city", "weather specification", "weekday" and "hijri date". In addition, the data contains a list of entities' synonyms. Table 6 and Table 7 present, respectively, some examples of intents and entities and IPA responses.

The main function of the Tunisian IPA is to provide answers to several inquiries about the weather, time, date and email box. We have prepared several possible stories that simulate a real conversation between a user and an IPA. A story is a representation of a conversation between a user and an IPA transformed into a particular format. The user request is expressed as intent (entities when necessary) and the assistant's responses and actions are expressed as action names [80]. Stories are used to train models that are able to generalize to unseen conversation paths. We identified 24 possible stories for requesting services in Tunisian. Figure 3 presents an example of a story. It is composed of a set of user requests (i.e., intent: greet, intent: ask_email and intent: thanks) and actions which the IPA should do (i.e., action: utter_start, action: action_mail, action: utter_thanks.). In a story, we mark entities which the IPA should

identify and save. The attribute "slot_was_set" is used to this end. Table 8 presents a real example of the story presented in Figure 3. For generating the suitable response for some intents (i.e., provide_weather_forecasts, ask_mail and ask_time_date), we have extracted the suitable information from three APIs: Accuweather API[10], google_api_python_client[11] and ummalqura.hijri_date API[12].

```
pipeline:
  - name: WhitespaceTokenizer
  - name: RegexFeaturizer
  - name: LexicalSyntacticFeaturizer
  - name: CountVectorsFeaturizer
  - name: CountVectorsFeaturizer
    analyzer: char_wb
    min_ngram: 1
    max_ngram: 4
  - name: DIETClassifier
    epochs: 100
    constrain_similarities: true
    model_confidence: linear_norm
  - name: EntitySynonymMapper
  - name: ResponseSelector
    epochs: 100
    constrain_similarities: true
  - name: FallbackClassifier
    threshold: 0.3
    ambiguity_threshold: 0.1
```

```
policies:
  - name: MemoizationPolicy
  - name: TEDPolicy
    max_history: 5
    epochs: 100
    constrain_similarities: true
  - name: RulePolicy
```

Figure 3. The pre-configured RASA pipeline and polices.

Table 6. Examples of intents and entities.

Intent	Example	Entity
greet	عالسلامة (çAlslAmħ) '*Hello*'	-
affirm	باهي (bAhy) '*Ok*'	-
goodbye	بالسلامة (bAlslAmħ) '*Good bye*'	-
thanks	صحيت (SHyt) '*Well-done'*	-
provide_weather_forecasts	بالله شنوة احوال الطقس؟ (bAllh šnwħ AHwAl AlTqs) *How is the weather?*	-
	شنية احوال الطقس في الجمّ المهدية (šnyħ AHwAl AlTqs fy Aljm~ Almhdyħ) '*How is the weather in Djem Mahdia?*'	[المهدية] {'entity': 'tun_city', 'value':'Mahdia'} [الجمّ]{'entity': 'tun_city', 'value': 'Mahdia'}
ask_mail	أقرا لي ليزمايل متاعي (ÂqrA ly lyzmAyl mtAçy) '*Please read my emails*'	[ليزمايل]{'entity': 'mail', 'value': 'mail'}
ask_date	اليوم في قداه(Alywm fy qdAh) '*What is the date of today?*'	[اليوم]{'entity':'date', 'role':'Day'}
	أحنا قداش في العام الهجري؟ (ÂHnA qdAš fy AlçAm Alhjry?) '*What is the Hijri date today?*'	[العام الهجري]{'entity':'date', 'role':'Year_Hejri'}

4.2.2 Dataset

To use RASA NLU, we prepared a training dataset to recognize intents and extract entities. The training data includes about 722 sentences with marked entities to train the RASA NLU. More specifically, there are 84.63% of sentences with entities which are presented according to weather specifications, date, time and email. The training data contains both interrogative and declarative sentences. We have also defined 46 synonyms for several entities (e.g. Tunisian cities, wind, clouds, etc.). We have used the person's name lexicon, composed of 538 entries.

4.2.3 Evaluation

First, to evaluate the performance of our NLU module, we have measured the numbers of intents and

[10] http://dataservice.accuweather.com/forecasts/
[11] https://pypi.org/project/google-api-python-client/
[12] https://github.com/borni-dhifi/ummalqura

Table 7. Examples of pre-defined responses.

Response type	**Example**	**Signification**
utter_city_ identification	(ÂçTyny AlwlAyħ mtAçk) أعطيني الولاية متاعك *Give me the name of your state.*	City identification
utter_bye	بالسلامة (bAlslAmħ) '*Bye*'	Bye
utter_thanks	*'You are welcome,'* (mn γyr mzyħ) مزية من غير	Thanks
utter_start	(çAlslAmħ) {person_name} عالسلامة '*Hello {person_name}*'	Start the conversation
utter_start_first	عالسلامة آنا المساعد الشخصي متاعك. تنجم تعرفني بيك؟ شنو اسمك؟ (çAlslAmħ ĀnA AlmsAçd AlšxSy mtAçk. tnjm tçrfny byk? šnw Asmk?) '*Hello. I am your personal assistant. Can I recognize you? What is your name?*'	First conversation

```
- story: ask_mail_story_1
  steps:
  - intent: greet
  - action: utter_start
  - intent: ask_mail
    entities:
    - mail: mail
  - slot_was_set:
    - mail: mail
  - action: action_mail
  - intent: thanks
  - action: utter_thanks
```

Figure 4. Example of Tunisian dialect story.

entities correctly classified. Due to the small size of the collected corpus, we have applied 5-cross validation and 10-cross validation to evaluate our NLU module. We have calculated the accuracy, F1-score and precision measures. Table 9 shows that entity-extraction accuracy is generally good. The accuracy value is 0.97 for both 10- and 5-cross validation. The results show that the F1-score scales from 0.74 for cloud entity extraction to 1 for multiple entities (e.g. month name). The failure in the classification of some entities can be explained by the presence of some entities composed of two words or more (e.g. عام العربي (çAm Alçrby) '*Hejri Year'*). The DIET classifier [81] is not able to detect the whole components of an entity. Sometimes, it detects the first or second part of the entity. In other cases, it fails to detect all the parts. When it comes to intents, the accuracy is 0.951 for 10-cross validation and 0.947 for 5-cross validation (See Table 9). There are some intent-classification mistakes related to greeting, denying and bye intents. By analyzing errors, we observe that the classifier makes an error for closely related utterances like عالسلامة (çAlslAmħ) '*hello'* and بالسلامة (bAlslAmħ) '*bye*'. In addition, some Tunisians use the same utterances for greeting and good-bye (i.e., السلام عليكم (AlslAm çlykm) to say '*hello*' and '*goodbye*'). In our future work, to avoid some errors, we propose to apply some pre-processing steps (e.g. tokenization, parsing, base phrase chunking, etc.) to requests before classification steps.

Table 8. Example of conversation between the user and IPA according to the story presented in Fig. 3.

User	عالسلامة'*Hello'* (çAlslAmħ)
IPA	إيناس عالسلامة'*Hello Ines'* (çAlslAmħ ĂynAs)
User	تشوف لي عنديشي مايل جديد (tšwf ly çndyšy mAyl jdyd) *'Can you tell me if I have new email?'*
IPA	عندك زوز مايلوات جدد *'You have two new emails.'* (çndk zwz mAylwAt jdd)
User	يعيشك مرسي'*Thank you'* (myrsy yçyšk)
IPA	(mn γyr mzyħ) *'You are welcome,'* من غير مزية

Moreover, to evaluate the quality of a full-dialog system; namely, NLU, DM and NLG modules, we have evaluated dialogs end-to-end by running through test stories. For this purpose, we used 15 stories. We obtained an accuracy of about 60%. Some errors are related to misclassification of some intents and/or entities. We have also evaluated the action level of the RASA core. The action-level results

Table 9. Entities' and intents' classification results.

	Intents		Entities	
	5-cross validation	**10-cross validation**	**5-cross validation**	**10-cross validation**
Accuracy	0.947	0.951	0.974	0.971
F1-score	0.945	0.95	0.966	0.966
Precision	0.951	0.954	0.977	0.978

measure the numbers for each intent-entity extraction prediction in all of the test stories. We obtained the following results: 0.899, 0.894 and 0.915, respectively, for F1-score, precision and accuracy.

4.3 Speech Synthesis

4.3.1 Proposed Method

End-to-end neural network architectures are widely used in many SS tasks. Unlike pipeline-based techniques, they are structured as a single component. End-to-end architectures learn all the steps between the initial input phase and the final output result and generate a single model. They reduce the need for expensive domain expertise and arduous feature engineering and require only minimal human annotation [64]. Among the famous and successful proposed end-to-end architectures proposed for SS, we cite Tacotron 2 [69]. Tacotron 2 is composed of two components: a sequence-to-sequence architecture spectrogram prediction network with attention and a flow-based implementation of WaveGlow [64]. For TA, we applied the Tacotron 2 architecture, which was updated by [64] in order to synthesize MSA. According to [64], the sequence-to-sequence spectrogram consists of an encoder and a decoder. The encoder takes a phonetized text as input and produces a hidden feature vector representation, which goes to the decoder and generates the mel-spectrograms of the given input characters. Then, the spectrograms are passed to a five-layer post-net. Finally, the WaveGlow vocoder, a flow-based generative network, is trained alongside using the mel-spectrograms and generates the voice as the output. We have used the open-source phonetization algorithm proposed by Nawar Halabi[13] to phonetize the input text.

4.3.2 Dataset

As the first attempt for our SS model, we have decided to train our model using one speaker transcriptions. Hence, we have trained Tacotron 2 on TA transcriptions, which contain about 1 hour and 33 minutes of speech composed of 2180 utterances. The corpus is composed of a pair of audio files and their transcriptions. We collected text from some Tunisian stories and some chapters from the Tunisian constitution in TA. We have divided them into short sentences which, then, have been recorded by a Tunisian woman (native speaker) in a silent environment. We manually recorded speech audio files using Audacity software[14]. We have used the Buckwalter transliteration[15] for the input text. Due to the unavailability of diacritization system for the TA and the slowness of manual transcription, we have decided to ignore all diacritics. Like SR corpus, we have converted files into wave format with a mono audio channel with a sample rate of 22050 Hz in order to be read by the Tacotron 2 pipeline. This dataset is used for training and validating the model. For testing our model, we have used a set composed of 2445 utterances, which consists of possible responses that the TA IPA can return to the user. The utterances include greetings, bye, weather forecasts, as well as time and date information.

4.3.3 Evaluation

Qualitative analysis was realized by using human ratings. We have calculated the subjective Mean Opinion Score (MOS), a rating of how good the synthesized utterances are for audio naturalness and comprehensiveness. Each utterance is evaluated by two raters. A score ranging from 1 to 5 was given to each utterance. 1 is given to bad audio, while 5 is given to the most natural audio. Table 10 presents the evaluation results. We have obtained an average MOS of 3.08. The score is encouraging for a first attempt to generate an SS model for TA. It shows that our SS can generate voice (the output of the IPA), which is almost natural and understandable. The analysis of the SS output shows that the majority of

[13] https://github.com/nawarhalabi/Arabic-Phonetiser
[14] https://www.audacityteam.org/
[15] http://www.qamus.org/transliteration.htm

mistakes are related to some phonemes that our model is not able to pronounce. Also, it is not able to synthesize some words. We can explain this failure by the absence of some phonemes in the training corpus. For example, the word ثلتطاش (θltTAš) '*thirteen*' is mispronounced due to the absence of the phoneme related to the letter š ش. We remark that our MOS score is lower than the [64] score. First, they used a diacritized corpus. The presence of short vowels helps learn the pronunciation. Also, their corpus is relatively bigger than ours.

Table 10. Evaluation results

Raters	MOS average
1	3.32
2	2.84
Average	**3.08**
MSA Tacotron 2 model [64]	**4.21**

5. CONCLUSION AND FUTURE WORK

In this paper, we present an attempt to create an Intelligent Personal Assistant (IPA) for the Tunisian dialect. We studied different approaches proposed for developing a dialog system, in particular, a task-oriented system. We prepared the basic components of an IPA: speech components (speech recognition model and speech synthesis model) and a dialog system core (natural-language understanding, Dialog management and language generation). We have applied deep-learning techniques: CNN [1], RNN encoder-decoder [2] and end-to-end approaches for creating IPA speech components (Speech Recognition and Speech Synthesis). In addition, we have explored the available and free dialog platform for understanding and generating the suitable response in TA for a request. Despite the lack of TA resources and as the first attempt, the evaluation results of some components are acceptable. We have proved the feasibility of creating an IPA with free resources while the language is under-resourced.

For future work, we have two main objectives. First, we intend to improve the quality of the proposed components. We intend to expand the size of all corpora by adding code-switching utterances and test other deep architectures. For speech components, we plan to add diacritics for our transcriptions. Their roles are important for the Arabic language. For speech recognition, we also plan to add more speakers to our corpus in order to recognize different speakers. We aim to augment the size of the corpus for the core of the dialog system by adding more services to the IPA. We intend to apply Transformer [82] in order to build some TA NLP tools and integrate them into the Rasa pipeline. The model will be able in the future to detect more complex entities and perform more complicated tasks. The second objective is the integration and testing of the developed components in the open-source vocal IPA, "openjarvis". It is designed to be executed on an energy-saving system, like the Raspberry Pi. It is a customizable IPA and the integration of new components is relatively easy.

REFERENCES

[1] K. O'Shea and R. Nash, "An Introduction to Convolutional Neural Networks," CoRR, vol. abs/1511.0, 2015.

[2] A. Hannun et al., "Deep Speech : Scaling up End-to-end Speech Recognition," arXiv1412.5567v2 [cs. CL], pp. 1–12, 2014.

[3] I. Lopatovska, "Overview of the Intelligent Personal Assistants," Ukr. J. Libr. Inf. Sci., no. 3, pp. 72–79, DOI: 10.31866/2616-7654.3.2019.169669, 2019.

[4] K. Jokinen and M. McTear, "Spoken Dialogue Systems," Synthesis Lectures on Human Lang. Technolo., Synthesis., Morgan & Claypool Publishers, DOI: 10.2200/S00204ED1V01Y200910HLT005, 2010.

[5] N. Goksel-Canbek and M. E. Mutlu, "On the Track of Artificial Intelligence: Learning with Intelligent Personal Assistants," Int. J. Hum. Sci., vol. 13, no. 1, pp. 592–601, DOI: 10.14687/ijhs.v13i1.3549, 2016.

[6] A. V. Román, D. P. Martínez, Á. L. Murciego, D. M. Jiménez-Bravo and J. F. de Paz, "Voice Assistant Application for Avoiding Sedentarism in Elderly People Based on IoT Technologies," Electronics, vol. 10, no. 980, 2021.

[7] Y. Matsuyama, A. Bhardwaj, R. Zhao, O. J. Romero, S. A. Akoju and J. Cassell, "Socially-aware Animated Intelligent Personal Assistant Agent," Proc. of the 17th Annual Meeting in Special Interest Group on Discourse and Dialogue (SIGDIAL 2016) , pp. 224–227, DOI: 10.18653/v1/w16-3628, 2016.

[8] J. Santos, J. J. P. C. Rodrigues, B. M. C. Silva, J. Casal, K. Saleem and V. Denisov, "An IoT-based Mobile Gateway for Intelligent Personal Assistants on Mobile Health Environments," J. Netw. Comput. Appl., vol. 71, pp. 194–204, DOI: 10.1016/j.jnca.2016.03.014, 2016.

[9] M. T. Talacio, Development of an Intelligent Personal Assistant to Empower Operators in Industry 4.0 Environments, M.Sc. Thesis, School of Technology and Management of Bragança. University of Bragança, 2020.

[10] E. Balcı, "Overview of Intelligent Personal Assistants," Acta INFOLOGICA, vol. 3, no. 1, pp. 22–33, DOI: 10.26650/acin.454522, 2019.

[11] K. Zdanowski, "Language Support in Voice Assistants Compared," Summa Linguae Technologies, Accessed on: Aug. 01, 2022, [Online], Available: https://summalinguae.com/language-technology/language-support-voice-assistants-compared/, 2021.

[12] I. Zribi, R. Boujelbane, A. Masmoudi, M. Ellouze, L. Belguith and N. Habash, "A Conventional Orthography for Tunisian Arabic," Proc. of the 9th International Conference on Language Resources and Evaluation (LREC'14), vol. Proc., pp. 2355–2361, Reykjavik, Iceland, 2014.

[13] A. Bouzemni, "Linguistic Situation in Tunisia: French and Arabic code switching," INTERLINGUISTICA, pp. 217–223, 2005.

[14] I. Zribi, M. Ellouze, L. H. Belguith and P. Blache, "Spoken Tunisian Arabic Corpus 'STAC': Transcription and Annotation," Resarch in Computing Science, vol. 90, pp. 123–135, 2015.

[15] I. Zribi, M. Ellouze, L. H. Belguith and P. Blache, "Morphological Disambiguation of Tunisian Dialect," J. King Saud Univ.-Comput. Inf. Sci., vol. 29, no. 2, pp. 147–155, 2017.

[16] H. Chen, X. Liu, D. Yin and J. Tang, "A Survey on Dialogue Systems: Recent Advances and New Frontiers," arXiv:1711.01731v3, no. 1, 2018.

[17] H. B. Hashemi, A. Asiaee and R. Kraft, "Query Intent Detection Using Convolutional Neural Networks," WSDM QRUMS 2016 Workshop, DOI: 10.1145/1235, 2016.

[18] K. Sreelakshmi, P. C. Rafeeque, S. Sreetha and E. S. Gayathri, "Deep Bi-directional LSTM Network for Query Intent Detection," Procedia Computer Science, vol. 143, pp. 939–946, 2018.

[19] A. Deoras and R. Sarikaya, "Deep Belief Network Based Semantic Taggers for Spoken Language Understanding," Proc. Interspeech 2013, pp. 2713-2717, DOI: 10.21437/Interspeech.2013-623, 2013.

[20] P. S. Huang, X. He, J. Gao, L. Deng, A. Acero and L. Heck, "Learning Deep Structured Semantic Models for Web Search Using Clickthrough Data," Proc. of the 22nd ACM Int. Conf. on Information & Knowledge Management (CIKM '13), pp. 2333–2338, DOI: 10.1145/2505515.2505665, 2013.

[21] W. A. Abro, A. Aicher, N. Rachb, S. Ultes, W. Minker and G. Qi, "Natural Language Understanding for Argumentative Dialogue Systems in the Opinion Building Domain," Knowledge-Based Syst., vol. 242, DOI: 10.1016/j.knosys.2022.108318, 2022.

[22] J. D. Williams, "Web-style Ranking and SLU Combination for Dialog State Tracking," Proc. of the 15th Annu. Meet. Spec. Interes. Gr. Discourse Dialogue (SIGDIAL 2014), pp. 282–291, DOI: 10.3115/v1/w14-4339, 2014.

[23] S. Sharma, P. K. Choubey and R. Huang, "Improving Dialogue State Tracking by Discerning the Relevant Context," Proc. of the Conf. of the North American Chapter of the Association for Computational Linguistics: Human Lang. Technolog. (NAACL HLT 2019), vol. 1, DOI: 10.18653/v1/n19-1057, 2019.

[24] Q. Xie, K. Sun, S. Zhu, L. Chen and K. Yu, "Recurrent Polynomial Network for Dialogue State Tracking with Mismatched Semantic Parsers," Proc. of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue, pp. 295–304, DOI: 10.18653/v1/w15-4641, Prague, Czech Republic, 2015.

[25] Z. Yan, N. Duan, P. Chen, M. Zhou, J. Zhou and Z. Li, "Building Task-oriented Dialogue Systems for Online Shopping," Proc. of the 31st AAAI Conf. on Artificial Intell. (AAAI-17), pp. 4618-4625, 2017.

[26] H. Cuayáhuitl, S. Keizer and O. Lemon, "Strategic Dialogue Management *via* Deep Reinforcement Learning," arXiv:1511.08099v1, pp. 1–10, 2015.

[27] A. Stent, R. Prasad and M. Walker, "Trainable Sentence Planning for Complex Information Presentation in Spoken Dialog Systems," Proc. of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04), pp. 79-86, DOI: 10.3115/1218955.1218966, Barcelona, Spain, 2004.

[28] T. H. Wen et al., "Stochastic Language Generation in Dialogue Using Recurrent Neural Networks with Convolutional Sentence Reranking," Proc. of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue, pp. 275-284, DOI: 10.18653/v1/w15-4639, Prague, Czech Republic, 2015.

[29] T. H. Wen, M. Gašić, N. Mrkšić, P. H. Su, D. Vandyke and S. Young, "Semantically Conditioned LSTM-based Natural Language Generation for Spoken Dialogue Systems," Proc. of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 1711–1721, DOI: 10.18653/v1/d15-1199, 2015.

[30] H. Zhou, M. Huang and X. Zhu, "Context-aware Natural Language Generation for Spoken Dialogue Systems," Proc. of the 26th Int. Conf. on Computational Linguistics: Technical Papers, pp. 2032–2041, Osaka, Japan, 2016.

[31] O. Dušek and F. Jurcicek, "Sequence-to-sequence Generation for Spoken Dialogue *via* Deep Syntax Trees

and Strings," Proc. of the 54th Annual Meeting of the Association for Computational Linguistics, vol. 2: Short Papers, pp. 45-51, DOI: 10.18653/v1/p16-2008, Berlin, Germany, 2016.

[32] T. H. Wen and S. Young, "Recurrent Neural Network Language Generation for Spoken Dialogue Systems," Computer Speech & Language, vol. 63, DOI: 10.1016/j.csl.2019.06.008, 2020.

[33] T. H. Wen et al., "A Network-based End-to-end Trainable Task-oriented Dialogue System," Proc. of the 15th Conf. of the European Chapter of the Association for Computational Linguistics, vol. 1: Long Papers, pp. 438–449, Valencia, Spain, April 3-7, 2017.

[34] A. Bordes, Y. Lan Boureau and J. Weston, "Learning End-to-end Goal-oriented Dialog," Proc. of the 5th Int. Conf. Learn. Represent. (ICLR 2017), 2017.

[35] C. Li, L. Li and J. Qi, "A Self-attentive Model with Gate Mechanism for Spoken Language Understanding," Proc. of the 2018 Conference on Empirical Methods in Natural Language Processing, pp. 3824–3833. DOI: 10.18653/v1/D18-1417, 2018.

[36] T. Bocklisch, J. Faulkner, N. Pawlowski and A. Nichol, "Rasa: Open Source Language Understanding and Dialogue Management," Proc. of NIPS 2017 Conversational AI Workshop, pp. 1–9, Long Beach, USA, 2017.

[37] B. A. Shawar, "A Chatbot as a Natural Web Interface to Arabic Web QA," Int. J. Emerg. Technol. Learn. (iJET), vol. 6, no. 1, pp. 37-43, DOI: 10.3991/ijet.v6i1.1502, 2011.

[38] S. M. Yassin and M. Z. Khan, "SeerahBot: An Arabic Chatbot about Prophet's Biography," Int. J. Innov. Res. Comput. Sci. Technol. (IJIRCST), vol. 9, no. 2, DOI: 10.21276/ijircst.2021.9.2.13, 2021.

[39] D. Abu Ali and N. Habash, "Botta : An Arabic Dialect Chatbot," Proc. of the 26th Int. Conf. on Comput. Linguist.: Sys. Demonstrat. (COLING 2016), pp. 208–212, Osaka, Japn, 2016.

[40] D. Al-ghadhban and N. Al-twairesh, "Nabiha : An Arabic Dialect Chatbot," Int. J. of Advanced Computer Sci. and App. (IJACSA) vol. 11, no. 3, pp. 452–459, 2020.

[41] A. A. Abdelhamid, H. Alsayadi, I. Hegazy and Z. T. Fayed, "End-to-end Arabic Speech Recognition: A Review," Proc. of the 19th Conf. of Language Engineering (ESOLEC'19), Bibliotheca Alexandrina, 2020.

[42] A. M. Dammak, "Approche Hybride Pour la Reconnaissance Automatique de la Parole Pour la Langue Arabe," Environnements Informatiques pour l'Apprentissage Humain, Université du Maine, Français, ⟨NNT : 2016LEMA1040⟩, 2016.

[43] S. Dua et al., "Developing a Speech Recognition System for Recognizing Tonal Speech Signals Using a Convolutional Neural Network," Appl. Sci., vol. 12, no. 12, p. 6223, DOI: 10.3390/app12126223, 2022.

[44] A. Y. Hannun, D. Jurafsky, A. L. Maas and A. Y. Ng, "First-pass Large Vocabulary Continuous Speech Recognition Using Bi-directional Recurrent DNNs," arXiv 1408 . 2873v2 [cs . CL], pp. 1–7, 2014.

[45] Y. Peng and K. Kao, "Speech to Text System: Pastor Wang Mandarin Bible Teachings (Speech Recognition)," CS230: Deep Learning, Stanford Univ., CA., 2020.

[46] N. Zeghidour et al., "Fully Convolutional Speech Recognition," arXiv:1812.06864v2, pp. 25–29, 2019.

[47] A. Agarwal and T. Zesch, "German End-to-end Speech Recognition Based on DeepSpeech," Proc. of the 15th Conf. on Natural Language Processing (KONVENS 2019), pp. 111-119, 2019.

[48] V. Pratap et al., "Wav2Letter++: The Fastest Open-source Speech Recognition System," Proc. of the 2019 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP), pp. 2–6, Brighton, UK, 2018.

[49] S. Qin, L. Wang, S. Li, J. Dang and L. Pan, "Improving Low-resource Tibetan End-to-end ASR by Multilingual and Multilevel Unit Modeling," Eurasip J. Audio, Speech, Music Process., vol. 2022, no. 1, DOI: 10.1186/s13636-021-00233-4, 2022.

[50] L. Lamel and J. Gauvain, "Automatic Speech-to-text Transcription in Arabic," ACM Transactions on Asian Language Information Processing, vol. 8, no. 4, DOI: 10.1145/1644879.1644885, 2009.

[51] M. Elshafei and H. Al-Muhtaseb, "Speaker-independent Natural Arabic Speech Recognition System," Proc. of the Int. Conf. on Intelligent Systems., [Online], Available: https://www.researchgate.net/publication/303873329_Natural_speaker_independent_arabic_speech_recognition_system_based_on_HMM_using_sphinx_tools, 2010.

[52] A. Ben Ltaief, Y. Estève, M. Graja and Lamia Hadrich Belguith, "Automatic Speech Recognition for Tunisian Dialect," Language Resources and Evaluation, vol. 52, no. 1, pp.249-267, DOI: 10.1007/s10579-017-9402-y, hal-01592416, 2018.

[53] A. Masmoudi, M. Ellouze Khmekhem, Y. Esteve, L. Hadrich Belguith and N. Habash, "A Corpus and Phonetic Dictionary for Tunisian Arabic Speech Recognition," Proc. of the 9th Int. Conf. Lang. Resour. Eval., vol. 3, no. 1, pp. 306–310, 2014.

[54] A. Messaoudi, H. Haddad, C. Fourati et al., "Tunisian Dialectal End-to-end Speech Recognition Based on DeepSpeech," Procedia Comput. Sci., vol. 189, pp. 183–190, DOI: 10.1016/j.procs.2021.05.082, 2021.

[55] S. N. Kayte, M. Mundada, S. Gaikwad and B. Gawali, "Performance Evaluation of Speech Synthesis Techniques for English Language," Adv. Intell. Syst. Comput., vol. 439, no. June, pp. 253–262, 2016.

[56] C. Quillen, "Autoregressive HMM Speech Synthesis," Proc. of the 2012 IEEE Int.l Conf. on Acoustics, Speech and Signal Processing (ICASSP), DOI: 10.1109/ICASSP.2012.6288800, Kyoto, Japan, 2012.
[57] M. Shannon and W. Byrne, "Autoregressive HMMs for Speech Synthesis," Proc. of the 10th Int. Conf. of the Int. Speech Comm. Associa. (Interspeech 2009), DOI: 10.21437/interspeech.2009-135, 2009.
[58] S. Roekhaut, S. Brognaux, R. Beaufort and T. Dutoit, "eLite-HTS: A NLP Tool for French HMM-based Speech Synthesis," Proc. Annu. Conf. Int. Speech Commun. Assoc. (Interspeech 2014), Singapore, 2014.
[59] S. Le Maguer, N. Barbot and O. Boeffard, "Evaluation of Contextual Descriptors for HMM-based Speech Synthesis in French," Proc. of the 8th ISCA Work, Speech Synth., HAL Id: hal-00987809, version 1, 2013.
[60] K. M. Khalil and C. Adnan, "Arabic Speech Synthesis Based on HMM," Proc. of the 15th IEEE Int. Multi-Conf. on Systems, Sig. & Devic. (SSD), DOI: 10.1109/SSD.2018.8570388, Hammamet, Tunisia, 2018.
[61] A. Amrouche, A. Abed and L. Falek, "Arabic Speech Synthesis System Based on HMM," Proc. of the 6th IEEE Int. Conf. on Electrical and Electronics Eng. (ICEEE), DOI: 10.1109/ICEEE2019.2019.0022, Istanbul, Turkey, 2019.
[62] H. Al Masri and M. E. Za'ter, "Arabic Text-to-speech (TTS) Data Preparation," arXiv:2204.03255v1, [Online], Available: http://arxiv.org/abs/2204.03255, 2022.
[63] K. M. Khalil and C. Adnan, "Arabic HMM-based Speech Synthesis," Proc. of the IEEE 2013 Int. Conf. on Electri. Eng. and Soft. Appl., DOI: 10.1109/ICEESA.2013.6578437, Hammamet, Tunisia, 2013.
[64] F. K. Fahmy, M. I. Khalil and H. M. Abbas, "A Transfer Learning End-to-end Arabic Text-to-speech (TTS) Deep Architecture, " arXiv:2007.11541v1 [eess.AS], 2020.
[65] A. van den Oord et al., "WaveNet: A Generative Model for Raw Audio Based on PixelCNN Architecture," arXiv:1609.03499, 2016.
[66] S. Arik et al., "Deep Voice: Real-time Neural Text-to-speech," Proc. of the 34th Int. Conf. Mach. Learn. (ICML 2017), vol. 1, no. Icml, pp. 264–273, 2017.
[67] Y. Wang et al., "Tacotron: Towards End-to-end Speech Synthesis," arXiv:1703.10135v2, pp. 1–10, 2017.
[68] I. Sutskever, O. Vinyals and Q. V. Le, "Sequence to Sequence Learning with Neural Networks," arXiv:1409.3215, 2014.
[69] J. Shen et al., "Natural TTS Synthesis by Conditioning Wavenet on MEL Spectrogram Predictions," DOI: 10.1109/ICASSP.2018.8461368, Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 2018.
[70] J. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho and Y. Bengio, "Attention-based Models for Speech Recognition," arXiv:1506.07503, 2015.
[71] I. Hadj Ali, Z. Mnasri and Z. Lachiri, "DNN-based Grapheme-to-phoneme Conversion for Arabic Text-to-speech Synthesis," Int. J. Speech Technol., vol. 23, pp. 569–584, DOI: 10.1007/s10772-020-09750-7, 2020.
[72] A. Abdelali, N. Durrani, C. Demiroglu, F. Dalvi, H. Mubarak and K. Darwish, "NatiQ: An End-to-end Text-to-speech System for Arabic," arXiv:2206.07373v1, 2022.
[73] N. Li, S. Liu, Y. Liu et al., "Neural Speech Synthesis with Transformer Network," Proc. of the 33rd AAAI Conf. on Artificial Intelligence (AAAI-19), pp. 6706–6713. DOI: 10.1609/aaai.v33i01.33016706, 2019.
[74] D. Amodei et al., "Deep Speech 2 : End-to-End Speech Recognition in English and Mandarin," arXiv 1512 . 02595v1 [cs . CL], pp. 1–28, 2015.
[75] A. Graves, S. Fernandez, F. Gomez and J. Schmidhuber, "Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks," Proc. of the 23rd Int. Conf. on Machine Learning (ICML '06), pp. 369-376, DOI: 10.1145/1143844.1143891, 2006.
[76] K. Heafield, "KenLM: Faster and Smaller Language Model Queries," Proc. of the 6th Workshop on Statistical Machine Translation, pp. 187–197, Edinburgh, Scotland, 2011.
[77] A. Mekki, I. Zribi, M. Ellouze and L. H. Belguith, "Sentence Boundary Detection of Various Forms of Tunisian Arabic," Language Resources and Evaluation, vol. 56, pp. 357–385, DOI: 10.1007/s10579-021-09538-4, 2022.
[78] N. Thi, M. Trang and M. Shcherbakov, "Enhancing Rasa NLU Model for Vietnamese Chatbot," Int. J. of Open Information Technologies (INJOIT), vol. 9, no. 1, pp. 31–36, 2021.
[79] Y. Windiatmoko, A. F. Hidayatullah and R. Rahmadi, "Developing FB Chatbot Based on Deep Learning Using RASA Framework for University Enquiries," CoRR, vol. abs/2009.1, [Online], Available: https://arxiv.org/abs/2009.12341, 2020.
[80] V. Vlasov, J. E. M. Mosig and A. Nichol, "Rasa Open Source Documentation," RASA DOCS, [Online], available: https://rasa.com/docs/rasa/, 2022.
[81] T. Bunk et al., "DIET: Lightweight Language Understanding for Dialogue Systems," arXiv:2004.09936v3, [Online], Available: https://arxiv.org/pdf/2004.09936.pdf, 2020.
[82] A. Chernyavskiy, D. Ilvovsky and P. Nakov, "Transformers: 'The End of History' for Natural Language Processing?," arXiv.2105.00813, [Online], Available: http://arxiv.org/abs/2105.00813, 2021.
[83] N. Habash, A. Soudi and T. Buckwalter, "On Arabic Transliteration," Arabic Computational Morphology, Part of the Text, Speech and Language Technology Book Series, vol. 38, pp. 15-22, 2007.
[84] S. Hussain, O. A. Sianaki and N. Ababneh, "A Survey on Conversational Agents," Proc. of the Workshops

"Towards Developing an Intelligent Personal Assistant for Tunisian Arabic", I. Zribi and L. H. Belguith.

of the 33rd Int. Conf. on Advanced Information Networking and Applications (Waina-2019), pp. 946-956, DOI: 10.1007/978-3-030-15035-8_93, Matsue, Japan, 2019.

ملخص البحث:

تـــمّ اقتـــراح أنظمـــةٍ ذكيـــةٍ علـــى نحـــوٍ مُكثَّـــف، بِـــدَفْع مـــن تقنيـــات الـــذّكاء الاصـــطناعي، لمســـاعدة النّـــاس فـــي أداء مهـــامّ متنوعـــة. ومـــن هـــذه الأنظمـــة الذّكيـــة المســـاعِد الشّخصـــي (IPA). فـــي هـــذه الورقـــة، نعـــرض محاولـــةً لإيجـــاد مســـاعِد شخصـــي ذكـــي يتفاعـــل مـــع المستخدِمين عبر اللهجة التّونسية، وهي اللهجة العامية المتداولة في تونس.

نقتـــرح ونطبّـــق طريقـــةً ســـهلة التّنفيـــذ لبنـــاء المكونـــات الأساســـية للمســـاعِد الشّخصـــي الـــذّكي بالعربيـــة التّونســـية، ونســـتخدم تقنيـــات الـــتّعلُّم العميـــق (CNN، والتّرميـــز-فـــكّ التّرميـــر RNN، وطريقـــة مـــن الطَّـــرَف الـــى الطَّـــرَف) مـــن أجـــل إيجـــاد مكونـــات المســـاعِد الشّخصـــي الذّكي، وهي: (تمييز الكلام، وتحليل الكلام).

بالاضـــافة الـــى ذلـــك، نستكشـــف مـــدى التـــوفر ومجانيـــة منصـــات الحـــوار لفهـــم وتوليـــد الإجابـــة المناســـبة للطلـــب باللهجـــة التونســـية. ولهـــذا الغـــرض، نقـــوم بإيجـــاد واســـتخدام نُصوصٍ باللّهجة العامّية المتداولة في تونس من أجل إنشاء النّماذج ذات العلاقة.

وقـــد أســـفر تقيـــيم النّظـــام المقتـــرح علـــى نتـــائج مشـــجّعة، اعتُبـــرت مقبولـــةً كمحاولـــة أولـــى تبْقى مُرشحةً للتّحسين والتّطوير في الأبحاث المستقبلية.