SPARTA: Speaker Profiling for ARabic TAlk

Wael Farhan[§], Muhy Eddin Za'ter[§], Qusai Abu Obaidah, Hisham al Bataineh, Zyad Sober, Hussein T. Al-Natsheh

Mawdoo3 Ltd

Amman, Jordan

{wael.farhan, muhy.zater, qusai.abuobaida, hisham.bataineh, zyad.sober, h.natsheh}@mawdoo3.com

[§]Equal contribution

Abstract— **This paper proposes a novel approach to an automatic estimation of three speaker traits from Arabic speech: gender, emotion, and dialect. After showing promising results on different text classification tasks, the multi-task learning (MTL) approach is used in this paper for Arabic speech classification tasks. The dataset was assembled from six publicly available datasets. First, The datasets were edited and thoroughly divided into train, development, and test sets (open to the public), and a benchmark was set for each task and dataset throughout the paper. Then, three different networks were explored: Long Short Term Memory (LSTM), Convolutional Neural Network (CNN), and Fully-Connected Neural Network (FCNN) on five different types of features: two raw features (MFCC and MEL) and three pre-trained vectors (i-vectors, d-vectors, and x-vectors). LSTM and CNN networks were implemented using raw features: MFCC and MEL, wher FCNN was explored on the pre-trained vectors while varying the hyper-parameters of these networks to obtain the best results for each dataset and task. MTL was evaluated against the single task learning (STL) approach for the three tasks and six datasets, in which the MTL and pre-trained vectors almost constantly outperformed STL. All the data and pre-trained models used in this paper are available and can be acquired by the public.**

I. INTRODUCTION

Speech is the basic form of human-to-human communication. Its signals hold valuable information about the speaker's identity, as they can convey the speaker's age, gender, emotional state, dialect, and other characteristics. Therefore, automatic detection of the speaker's features can inevitably enhance and affect various fields and has a wide range of applications, including, but not limited to, medical, commercial, forensics, and other real-world applications that involve human-machine interaction scenarios. Thus, intensive research endeavors recently started focusing on estimating and extracting important traits of speakers solely from speech.

The work presented in this paper concentrates on classifying the speaker's gender, dialect, and emotional tone. In most human-machine interactions, it is necessary that the gender of the speaker or user to be known. Emotion, on the other hand, is a fundamental factor in any human-to-human communication. The speaker's voice pitch, intonation, and rate, they all affect the emotions perceived by the listener and hence can change the semantics of the utterance, which in turn makes installing an effective emotion recognition system in human-machine interaction of great benefit. Moreover, recognizing the dialect of the speaker simplifies speech-related tasks such as Automatic

Speech Recognition (ASR), which makes the task of dialect identification an important part of speech recognition systems. The approach presented in this paper; Multi-task learning, has not been used to detect these traits of speakers from Arabic speech or any language in any previous work.

Although voice signals are mixed signals from which multiple features can be extracted simultaneously, until now, the majority of the voice-related recognition tasks pay more attention to the identification of a single feature from any voice utterance [1]. By contrast, few recognition systems can identify multiple features such as the speaker's gender and emotion simultaneously [2], [3]. The availability of speech recognition systems drastically decreases in the case of the Arabic language in line with the deficiency of available datasets.

In this paper, a novel multi-task learning framework is proposed for the detection of gender, emotion, and dialect simultaneously from Arabic speech, while also exploring various feature extraction methods and speech utterance modeling techniques. The dataset used to train and test the models were assembled and collected from existing datasets that can be acquired by the public. The division of datasets and their results can be used as a benchmark for future work (https://github.com/mawdoo3/sparta-benchmark). Finally, the method proposed in this paper outperforms the best result recorded in the literature on the tasks of one of the datasets. These tasks were selected as they help call centers and other industries to recognize their audience and the demand on these tasks are increasing.

The rest of the paper is structured as follows: section II presents related work in this area. Section III describes the collected datasets. Section IV demonstrates various feature extraction methods. The proposed method for learning multi-task classification is described in section V. Section VI presents the results obtained from the conducted experiments. Finally, section VII concludes the work presented in the paper.

II. RELATED WORK

The detection and estimation of the features and speaker's traits from speech have been a challenge for researchers and industries since the emergence of speech recognition [4], [5] due to multiple factors, including, but not limited to: background noise, different accents, and speaker dependence. However, research endeavors were able to propose alternative solutions and techniques to tackle this challenge.

A. Speaker Profiling

Researchers followed different approaches and explored various techniques for different speech recognition and detection tasks such as ASR, gender, emotion, and dialect detection [6], [7]. Previous work started with the use of raw and non-linguistic features such as Mel-Frequency Cepstral Coefficients (MFCCs), Linear Prediction Cepstral Coefficients (LPCCs) [1], energy and pitch, and building classifiers to estimate the speaker's traits from these features. Other work in the literature focused on improving utterance modeling in low dimensional vectors such as i-vector [8]–[10], x-vector [11], and d-vector [12]. These utterance modeling techniques showed significant enhancements in speaker verification and diarization tasks, hence researchers started to use them in speech classification tasks. The aforementioned features will be discussed in more detail later in this paper. Other various classifiers were also explored: Gaussian Mixture Models (GMM), Hidden Markov Models (HMM), Support Vector Machines (SVM), Neural Networks, and other hybrid classifiers or ensembles [13].

B. Arabic Speaker Profiling

A lack of available Arabic speech corpora for different speech recognition tasks, and with only a few efforts reported in the attempt of Arabic speech recognition in comparison with other global languages, alongside the fact that Arabic is a very phonetically rich language and significantly differs between dialects [14], [15]. All the mentioned facts acclimatize the challenge of speech recognition and detection tasks for the Arabic language. There have been some attempts for speech classification tasks either by building labeled speech corpora or by implementing and proposing classifiers for gender, emotion, dialect, or other detection tasks [13], [16], [17]. Examples of open-source data for Arabic Speech are Aljazeera that can be used for ASR, Qatar Computation, and Research Foundation Dialect Corpus. Other speech corpora can be acquired such as King Saud University speech corpus and emotion corpus.

The natural language processing (NLP) community has aggregated dialectal Arabic into five regional language groups: Egyptian (EGY), North African (NOR), Gulf or Arabian Peninsula (GLF), Levantine (LAV), and Modern Standard Arabic (MSA) [16]. The task of Arabic dialect detection solely from speech has been investigated by utilizing multiple approaches. Authors in [16] used SVM classifiers on top of i-vector-based utterances for the task of dialect detection. While other work in literature followed the approach of language identification [9], [16]. However, there have not been any reported attempts for Arabic dialect identification using multitask learning.

There are several categorizations for emotion in NLP. The most famous is Ekman's categorization [18], which has the following categories: anger, disgust, fear, happiness, sadness, and surprise. However, in this paper, the emotions are categorized following [19]'s categorization: anger, surprise, happiness, sadness, questioning, and neutral. Emotion detection from Arabic speech has been explored in previous work as in [2],

[13], where authors used multiple well-known classification methods and ensembles on raw features such as MFCC and pitch. Gender detection is an easier task in comparison with emotion and dialect detection in most languages including Arabic. The work in the literature performed well by building simple classifiers on raw features as in [13].

C. Multi-task Speaker Profiling

Multi-task learning is generally applied by sharing some hidden layers between all tasks while keeping several task-specific hidden and output layers, each with a loss function task [20], [21]. Researchers started scratching the surface in exploring multi-task learning for speech recognition tasks after its successful utilization in text-based classification tasks. It has first been applied to enhance the robustness and phoneme recognition for ASR systems [22]. However, this paradigm application has been extended to speech classification tasks. In [3], authors used multi-task learning for estimating age, height, weight, and smoking habits of speakers, using utterance modeling techniques and other classifiers. Researchers in [2] proposed a multi-task learning approach to simultaneously estimate speaker identity and the other two traits of the speaker, specifically, emotion and gender from Arabic speech.

This paper proposes the use of a multi-task learning paradigm to simultaneously estimate the gender, dialect, and emotion of the speaker, which slightly outperformed single task classifiers on the three aforementioned tasks. All the datasets and pre-trained models assembled, edited, and used in this work are available to the public (either as open source or can be acquired). Also, various utterance modeling techniques, features, and network architectures are explored throughout the experiments and are reported in this paper. Finally, the code used to pre-process and prepare the data is available to future researchers.

III. DATASETS

This section briefly describes the Arabic Speech datasets collected from the literature. The datasets were collected for emotion, dialect, and gender recognition tasks. It is worth mentioning that for this research, only datasets that are either open-source or can be acquired by the public were used.

Qatar Computing Research Institute (QCRI) Arabic dialect detection data: The training corpus was collected from the Broadcast News domain in four Arabic dialects: Egyptian, North African, Gulf, Levant, and Modern Standard Arabic (MSA) [16]. The recordings were segmented to avoid speaker overlap, and any non-speech parts such as music and background noise were removed. The speech corpus consists of 15,000 utterances, with around 300 utterances for each dialect, and another 1,500 utterances as a development set.

King Saud University Emotions (KSUEmotion) Corpus: This speech corpus contains 16 different sentences spoken with 6 distinct emotions: Neutral, happy, sad, angry, surprised, and questioning [19]. These 16 sentences are written in MSA

and spoken by 23 different speakers from Syria, Yemen, and Saudi Arabia. 10 of these speakers are male while the other 13 are female, which makes this data suitable for both emotion detection and gender detection tasks. This dataset has 3,280 utterances divided equally between genders, while each emotion category has 650 utterances except for questioning and anger having 320 utterances each.

Arabic Natural Audio Dataset (ANAD): This Arabic speech corpus was developed to recognize 3 types of emotions from Arabic speech: Happy, angry, and surprised [23]. Eight videos of live calls from online Arabic talk shows were downloaded and pre-processed by dividing the calls into turns: callers and receivers, and then silences, laughs, and noisy segments were removed. Finally, the utterances were labeled into happy, angry, and sad by 18 different Arabic listeners. However, while conducting the experiments, it was found that the duration of each segment was not long enough. To solve this, each segment was changed to have at least 3 seconds, and any silences were trimmed. The corpus after the new segmentation consists of 475 recordings.

Spoken Arabic Regional Archive (SARA): The contributors of this dataset collected from episodes and films published on YouTube played by native speakers with three different Arabic dialects and accents: Egyptian, the Arabian Peninsula dialect and Levantine dialect [24]. The dataset samples vary in length from 3 to 7 seconds in order to verify the minimum time required to detect the speaker's dialect. This corpus consists of 3,480 voice recordings divided evenly between the three dialects.

Multi Dialect Arabic Speech (MDAS): This corpus was designed to recognize three dialects along with Modern Standard Arabic (MSA). These dialects are Levant, Gulf, and Egyptian [25]. The recordings were conducted with the consent of 52 participants. The recordings' duration totaled around 32 hours. After the segmentation stage, 67,132 speech files were obtained.

King Saud University (KSU): King Saud University Arabic Speech Database was developed by Speech Group (SG) at King Saud University and consists of 590 hours of recorded Arabic speech from 269 males and females speakers [26]. The corpus was designed principally for speaker recognition research and other applications. However, in this research, it is used for gender detection. Only 49 hours were taken from the original corpus, which accounts for 9,050 utterances spoken by 350 different speakers.

All these datasets were aggregated together and then divided into train, development, and test sets while obeying to a couple of restrictions: (i) there should be no speaker overlap between the train, development, and test sets, (ii) each class of each dataset is almost divided between train(80%), development(10%) and test(10%) sets. Following these rules

implies that the entire dataset will be divided with the same percentages. When abiding with these restrictions, it is almost impossible to get a perfect split because each speaker has several utterances labeled with multiple classes. Choosing the set for a speaker could be optimal for one class, but not the other. A dataset split using this format can be reduced into an integer programming problem. Table I shows the distribution of each class in each set.

IV. Feature Extraction

This section briefly describes the features used to train the network. Five types of features are used in this paper: MFCC, MEL, i-vector, d-vector, and x-vector. They are briefly described below. It is worth mentioning that i-vector, d-vector, and x-vector extracted or trained during the experiments were either trained on data that is available to the public or extracted from available pre-trained models. Regarding the specification of the speech files, they were converted to the WAV file type, with a 16k sampling rate, mono-channel, and 16-bit precision.

A. Mel-spectrogram (Mel) Extraction

Mel-spectrogram and MFCC are common methods for feature extraction from speech. A speech signal is first segmented into frames, each with a length between 20 to 40 ms to get a stationary signal. The Discrete Fourier Transform (DFT) is calculated for each frame in order to calculate the power density. Then, the frequency response calculated from DFT is passed through triangular band-pass filters to smoothen the energy, which is then finally converted to the Mel-scale. In this paper, parameter selection follows [11] where frames are 25 ms each. FFT size is 1024, and the number of filters is 24.

B. Mel-Frequency Cepstral Coefficients (MFCC) Extraction

Previously, the steps to compute filter banks energies in a log-spaced frequency domain were discussed, which are known as Mel-spectrogram features. These steps were originally motivated by the way humans perceive audio. An additional step of passing the computed log-filter banks energies is through a Discrete Cosine Transform (DCT). Using DCT will de-correlate the filter banks energies and transform the frequency domain into time-like domain features. A $L = 14$ mel-scaled cepstral coefficients were calculated. The result coefficients are called Mel-Frequency Cepstral Coefficients (MFCC).

C. I-Vector Extraction

The need to represent speech utterances in a low dimensional, fixed-length vector regardless of the utterance length led to the emergence of i-vector-based frameworks [8]–[10] . I-vector captures the long-term characteristics of audio, such as the speaker's characteristics or recording device. Originally, this representation was developed for speaker recognition and verification, but it is now used in many areas of speech processing. Also, it has been widely used in the classification tasks in these papers, including dialect and emotion detection. In this paper, i-vectors of dimension 400 were trained on

TABLE I. DISTRIBUTION OF DATASETS

Dataset	Gender		Dialect					Emotion						Total
Labels	M	F	MSA	LEV	GUL	NOR	EGY	NEU	SAD	HAP	SUR	ANG	QUES	TOT
Train Set														
KSU	1,360	1,357	0	0	0	0	0	544	542	544	544	288	255	5,434
ANAD	0	0	0	0	0	0	0	0	0	122	33	210	0	365
QCRI	0	0	1,998	2,807	2,586	2,885	2,860	0	0	0	0	0	0	13,136
SARA	1,691	1,342	0	949	961	0	1,123	0	0	0	0	0	0	6,066
MDAS	0	0	12,910	7,746	12,910	0	20,655	0	0	0	0	0	0	54,221
KING	3,887	3,374	0	0	0	0	0	0	0	0	0	0	0	7,261
Total	6,938	6,073	14,908	11,502	16,457	2,885	24,638	544	542	666	577	498	255	86,483
Dev Set														
KSU	200	80	0	0	0	0	0	56	56	56	56	24	32	560
ANAD	0	0	0	0	0	0	0	0	0	19	4	32	0	55
QCRI	0	0	221	311	287	320	317	0	0	0	0	0	0	1,456
SARA	224	204	0	146	125	0	157	0	0	0	0	0	0	856
MDAS	0	0	1,291	1,291	1,290	0	2,582	0	0	0	0	0	0	6,454
KING	482	410	0	0	0	0	0	0	0	0	0	0	0	892
Total	906	694	1,512	1,748	1,702	320	3,056	56	56	75	60	56	32	10,273
Test Set														
KSU	79	200	0	0	0	0	0	56	55	56	56	24	32	558
ANAD	0	0	0	0	0	0	0	0	0	19	4	32	0	55
QCRI	0	0	283	348	265	355	315	0	0	0	0	0	0	1566
SARA	187	191	0	149	144	0	85	0	0	0	0	0	0	756
MDAS	0	0	1,291	1,291	1,291	0	2,582	0	0	0	0	0	0	6455
KING	478	419	0	0	0	0	0	0	0	0	0	0	0	897
Total	744	810	1,574	1,788	1700	355	2,982	56	55	75	60	56	32	10,287

the dataset mentioned in the previous section. I-vector is considered to be the first generation of low dimensional fixed-length vector representations of speech, as it is based on the factor analysis on Gaussian Mixture Model (GMM) mean supervectors and the Non-negative Factor Analysis (NFA) framework, which is based on a constrained factor analysis on GMM weights.

D. D-Vector Extraction

The d-vector features were extracted from a pre-trained network that had been trained on a text-independent speaker verification task using a large amount of data [12], [27]. The network input is 40-channel log-Mel spectrograms of a WAV of arbitrary length, passed to a stack of 3 LSTM layers of 768 cells, each followed by a projection to 256 dimensions. The final embedding is created by L2-normalizing the output of the top layer at the final frame. The output is an embedding vector of length 256 known as a d-vector.

E. X-Vector Extraction

X-vectors are fixed-dimensional embeddings extracted from a network trained on a speaker verification task [11]. Embeddings of this type need substantial amounts of data, which called for the need to augment the data by adding noise and reverberations as an inexpensive way to multiply the training data. Furthermore, it was found that x-vector benefited more from the augmentation techniques when compared to i-vector and performed better in speaker verification tasks. In this paper, x-vectors were extracted from a pre-trained model on large data.

All these features were pre-computed and stored into separate files, each containing the feature type for audio files in all datasets. This step was performed for two reasons: (i) to make the training process fast and efficient, (ii) to remove any indeterministic nature that could exhibit some of these extraction methods. We refrain from disclosing those files due to license agreements for some of the datasets which forbid redistribution.

V. METHOD

This section describes the multi-task learning networks implemented, trained, and their combinations. It also describes the tuned hyper-parameters and the training setup of the experiments.

A. Network Architecture

There are many approaches to build a network that tack-les the problem of multi-task speech classification. In our research, machine learning blocks that can be utilized in combination with specific input features to formulate an end-to-end prediction model are identified and briefly described. Fig. 1 illustrates the building blocks of the SPARTA network. The illustration does not represent a network, but a framework on which components can be connected to create a network. These building blocks are grouped into 3 main sections:

1) **Input Layer:** This layer was designed to extract feature representations from an utterance signal. These features, as described in the previous section, are either fixed-size learned feature extractions using methods
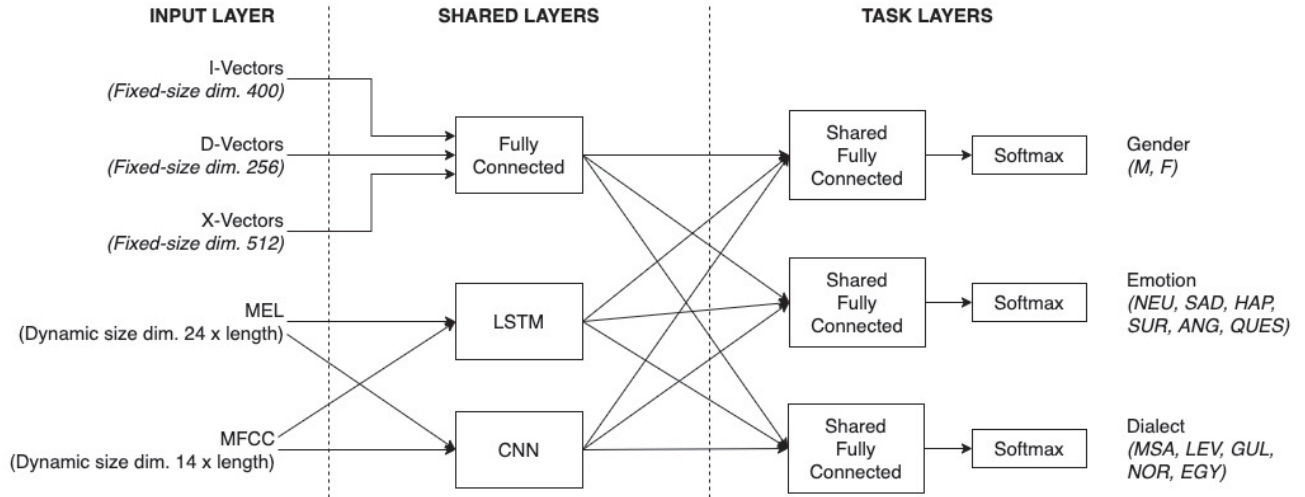
Fig. 1. Building blocks of SPARTA framework

such as i-vector, d-vector, and x-vector or dynamic-size frequency cepstrum representations such as MFCC and MEL. Given that fixed-size features learn different representations, concatenating these vectors could enrich the feature space. All possible permutations of vectors are generated: id-vector, ix-vector, dx-vector, and idx-vector. However, since MFCC is a compressed version of MEL, they were not concatenated, as having both would produce redundant information.

2) **Shared Layers:** These layers consume the output of the previous layer and transform it into an intermediate representation that captures more information to help classify the selected tasks. The type of shared layer chosen is dependent on the selected features. Fully connected layers are chosen to consume dense and fixed input representations (i, d, x vectors, and their combinations), while either convolutional layers or LSTM are chosen for variable-size inputs (MFCC and MEL). Certain combinations were omitted, such as consuming a fixed-size vector with an LSTM network which is designed to consume sequential variable length input. It is worth noting that the weights of these layers are tweaked by the error propagated from all tasks.

3) **Task Layers:** These layers are composed of several sub-networks where each sub-network consumes output from the shared layers and computes predictions for its assigned task. Each sub-network is built from a configurable number of fully connected layers, and ending with a softmax layer. Cross entropy loss is used across all tasks. Note that parameters are not shared between tasks, hence the back-propagation algorithm only tweaks one sub-network per batch. When performing single-task learning, only one prediction layer is active while the other two are disabled.

There are eleven possible variations available to connect the aforementioned blocks to build a multi-task speech classification model. All these models were built and trained with the dataset described in section III in both MLT and STL format.

B. Experimental Setup

As mentioned in the previous section, three network architectures were implemented followed by task-specific networks. The hyper-parameters of the networks were explored to obtain the best possible combination of parameters that results in the best accuracy and F1 score. Table II presents each hyper-parameter and their ranges.

In addition to exploring the network hyper-parameters, different training parameters and setups were investigated, including the optimizer, the learning rate and its decay, and the number of epochs. Furthermore, in order to balance the number of utterances for each task, a multiplication factor for the utterances of a selected task was added to ensure the balance of the datasets for all the tasks. The training setup has two variations: (i) train one task at a time wherein each epoch the model finishes training on one task then moves to the next, or (ii) shuffled where each batch is selected at random from any task.

A combination of manual and grid search algorithm [28] was applied to explore all tuning parameters. The model was evaluated against the dev set after each epoch in each training process. The best epoch's weights were stored, and the final scores were computed by evaluating the best model on the test set as seen in table III.

Finally, it is worth mentioning that the networks described in this section were implemented using the TensorFlow framework [29] and trained on GPUs of type Tesla T4.

VI. RESULTS AND DISCUSSION

To evaluate the multi-task learning paradigm against the conventional single-task learning for the aforementioned tasks, several experiments were conducted on the previously described networks with different features, while varying the hyper-parameters of the networks in an attempt to obtain the

TABLE II. Networks hyper-parameters

Network	Hyper-parameter	Range	Network	Hyper-parameter	Range
CNN	Features	MFCC, MEL	LSTM	Features	MFCC, MEL
	Number of filters	[32, 64]		Direction	Uni/bi- directional
	Filter size	[3, 4, 5]		Hidden nodes	[16, 32, 64, 128]
	Number of layers	[1, 2, 4]		Number of layers	[1,2,4]
	Activation	[relu, tanh]		Activation	[relu, tanh]
	Dropout	[0, 0.1, 0.2, 0.3]		Dropout	[0, 0.1, 0.2, 0.3]
Fully Connected	Features	i/d/x/id/ix/dx/idx-vector	Task Specific layers	Hidden nodes	[16, 32, 64, 128]
	Hidden nodes	[16, 32, 64, 128, 256]		Number of layers	[1,2]
	Number of layers	[1,2,4]		Activation	[relu, tanh, sigmoid]
	Activation	[relu, tanh, sigmoid]		Dropout	[0, 0.1, 0.2, 0.3, 0.5]
	Dropout	[0, 0.1, 0.2, 0.3]			

best possible results. Also, to create benchmarks for the used datasets, the results of each dataset separately are reported in the section.

A. Network Architecture Results

The best results reported for each network and features in terms of F1 scores are presented in Table III. From this table and throughout the experiments, it is evident that utterance modeling techniques such as i-vectors and d-vector outperform raw features (MFCC and MEL), especially in dialect and emotion detection tasks. This can be explained by the fact that these techniques take into consideration phonetic and lexical features with only a limited amount of data, unlike raw features that require far more hours of speech to capture such features on their own. Also, the performance resulting from the concatenation of i-vector, d-vector, and x-vector surpasses their individual performances, as each one of these captures different characteristics since they are trained on different and larger datasets. It is noteworthy that the performance of d-vector on its own is relatively poor. However, when concatenated with i-vector and x-vector the model's performance was boosted as it contains a different information type than the others and has been trained on different kinds of data. Furthermore, in the case of pre-trained vectors, multi-task learning outperformed single-task learning in each task. However, the difference was more apparent in the emotion and dialect detection tasks. To further indicate the importance of utterance modeling techniques, in comparison to MFCC and MEL, utterance modeling techniques achieved better results in the gender detection task. This can be attributed to the fact that these techniques were originally designed for speaker verification tasks, hence they can recognize the gender of the speaker better.

On the other hand, CNN and LSTM networks require more data to surpass the performance of fully connected networks with utterance modeling techniques. Also, in the case of CNN adn LSTM, multi-task learning slightly enhanced the performance of the model when compared to single-task learning, which can be attributed to the fact that the network had more data and more shared layers, which positively affected that network's performance. However, LSTM and CNN performed less in comparison to other networks, due to the networks degrading ability to capture learning in proportion to the length of the sequence.

B. Data Specific Results

Table IV illustrates the F1 score obtained for each dataset separately on single-task learning and multi-task learning (all datasets combined). This table sets a benchmark for all datasets used in this work. It can be noticed that the results of multi-task learning almost persistently outperform the single-task learning for each dataset, especially when using pre-trained vectors for utterances. The effect of multi-task learning enhancement is especially evident in the emotion and dialect detection tasks. However, on the gender tasks, the results are close between the single task learning and multi-task learning. The work in [2] used the KSUEmtotion dataset for emotion and gender recognition; the accuracy recorded was 79.3% for the task of emotion recognition and 98.7% on gender detection. The best experiment of the proposed model outperformed these results as shown in Table IV.

VII. Conclusion and Future Work

This paper presents a novel approach that utilizes multi-task learning for the tasks of gender, emotion, and dialect detection from Arabic speech. It also explores various pre-trained vectors, their concatenation, and raw features on different networks and implementations. Additionally, it introduces a new Arabic Speech benchmark collected from publicly available datasets with code to reproduce the features and pre-trained vectors.

Throughout the experiments conducted in this paper, the following can be concluded. First, pre-trained vectors such as i-vector, d-vector, and x-vector enhance any network's performance on detection tasks and should be further explored and improved on. Also, similar to text classification tasks, multi-task learning improves performance on speech classification and recognition tasks. Therefore, its utilization must be expanded further on speech-related tasks. Moreover, despite the significant shortage of publicly available corpora for Arabic speech, speech corpora that can be used for gender, emotion, and dialect detection tasks are assembled with benchmarks for each task and dataset. Future work of this paper should

TABLE III. RESULTS (F1 SCORES) FOR MULTI-TASK AND SINGLE-TASK FOR EACH TASK

Network	Features	Gender		Emotion		Dialect		Overall	
		STL	MTL	STL	MTL	STL	MTL	STL	MTL
Fully connected	d-vector	99.10	98.58	42.60	55	36.75	37.33	59.48	63.63
	i-vector	92.59	93.88	55.9	58.35	46.61	46.67	65.03	66.3
	x-vector	98.90	99.61	54.13	54.48	48.61	48.77	67.21	67.62
	id-vector	97.10	96.39	62.88	66.47	51.14	53.44	70.34	72.1
	ix-vector	**99.42**	99.48	62.26	61.98	52.33	52.59	71.41	71.35
	dx-vector	98.71	**99.81**	57.14	64.24	52.35	51.07	69.4	71.71
	idx-vector	99.10	99.41	**66.53**	**70.16**	**56.05**	**58.44**	**73.89**	**76.01**
CNN	MFCC	96.13	95.54	52.91	51.95	21.19	26.7	56.98	58.08
	MEL	97.16	95.54	51.18	51.25	22.74	27.89	57.02	58.22
LSTM	MFCC	96.39	95.49	54.9	57.04	26.8	29.07	59.3	60.53
	MEL	94.96	95.12	50.4	57.05	28	27.94	57.78	60.2

TABLE IV. RESULTS (F1 SCORES) FOR MULTI-TASK AND SINGLE-TASK FOR EACH EACH DATASET (G: GENDER, E: EMOTION, D: DIALECT)

Model	Features	Dataset							
		KSUEmotion G		KSUEmotion E		ANAD E		SARA G	
		ST	MT	ST	MT	ST	MT	ST	MT
CNN	MFCC	96.5	97.4	50.7	50.35	56.3	54.93	87.8	85.8
	MEL	92.9	94.1	50.51	50.13	51.11	50.81	83.8	87.1
LSTM	MFCC	80.1	98.25	51.9	55.53	32.3	44.64	85.2	87.39
	MEL	81.4	95.32	46.1	54.79	34.2	44.34	83	84.96
Fully Connected	d-vector	100	100	48.1	52.4	72.1	73.6	94.4	96.8
	i-vector	100	100	62.6	66.7	88.1	85.36	79.3	79.5
	x-vector	100	100	66.32	69.54	88.8	84.34	94.9	98.4
	id-vector	100	100	61.56	64.36	87.1	85.46	86.3	87.5
	ix-vector	100	100	**73.4**	75.4	90.3	91	98.4	97.8
	dx-vector	100	100	67	70.3	85.9	83.87	95.4	**99.4**
	idx-vector	**100**	**100**	72.34	**80.35**	**91.8**	**91.6**	**99.4**	97.6
		SARA D		KSU G		MDAS D		QCRI D	
		ST	MT	ST	MT	ST	MT	ST	MT
CNN	MFCC	26.6	25.9	98.9	98.8	24.2	23.9	25.8	25.2
	MEL	26.1	24.82	99.1	99.2	22.9	25.7	25.4	23.8
LSTM	MFCC	30.4	22.1	98.7	97.87	43.4	27.6	30.6	27.5
	MEL	28.8	21.2	98.3	97.13	30.7	25.91	31.0	28.49
Fully Connected	d-vector	40.1	36.32	**100**	99.6	40.7	38.36	34.7	29.1
	i-vector	44.7	44.67	98.6	97.9	47.5	45.68	52.3	50.93
	x-vector	38.3	37.93	99.8	100	56.7	53.7	38.5	37.5
	id-vector	55.4	50.3	99.6	99.3	55.9	53.21	50.7	50.8
	ix-vector	39.9	40.12	99.8	**100**	60	56.87	49.6	47.3
	dx-vector	31.2	30.45	99.8	99.8	55.9	54.3	39.4	36.8
	idx-vector	**50.4**	**49.8**	99.8	99.88	**60.6**	**70.67**	**53.6**	**54.46**

include enhancing pre-trained vectors for utterance modeling by training them on larger Arabic corpus and include available Arabic ASR data. Furthermore, explore more architectures and features such as self-attention, TDNN, and filter banks.

REFERENCES

[1] S. G. Koolagudi and K. S. Rao, "Emotion recognition from speech: a review," *International journal of speech technology*, vol. 15, no. 2, pp. 99–117, 2012.

[2] H. Chen, L. Xu, and Z. Yang, "Multi-dimensional speaker information recognition with multi-task neural network," in *2018 IEEE 4th International Conference on Computer and Communications (ICCC)*. IEEE, 2018, pp. 2064–2068.

[3] A. H. Poorjam, M. H. Bahari *et al.*, "Multitask speaker profiling for estimating age, height, weight and smoking habits from spontaneous telephone speech signals," in *2014 4th International Conference on*

Computer and Knowledge Engineering (ICCKE). IEEE, 2014, pp. 7–12.

[4] V. Radha and C. Vimala, "A review on speech recognition challenges and approaches," *doaj. org*, vol. 2, no. 1, pp. 1–7, 2012.

[5] N. Desai, K. Dhameliya, and V. Desai, "Feature extraction and classification techniques for speech recognition: A review," *International Journal of Emerging Technology and Advanced Engineering*, vol. 3, no. 12, pp. 367–371, 2013.

[6] D. Ververidis and C. Kotropoulos, "Automatic speech classification to five emotional states based on gender information," in *2004 12th European Signal Processing Conference*. IEEE, 2004, pp. 341–344.

[7] H. K. Palo, M. N. Mohanty, and M. Chandra, "Efficient feature combination techniques for emotional speech classification," *International journal of speech technology*, vol. 19, no. 1, pp. 135–150, 2016.

[8] W. Li, T. Fu, and J. Zhu, "An improved i-vector extraction algorithm for speaker verification," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2015, no. 1, p. 18, 2015.

[9] D. Martinez, O. Plchot, L. Burget, O. Glembek, and P. Matějka,

"Language recognition in ivectors space," in *Twelfth annual conference of the international speech communication association*, 2011.

[10] M. Senoussaoui, P. Kenny, N. Dehak, P. Dumouchel *et al.*, "An i-vector extractor suitable for speaker recognition with both microphone and telephone speech." in *Odyssey*, 2010, p. 6.

[11] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5329–5333.

[12] E. Variani, X. Lei, E. McDermott, I. L. Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 4052–4056.

[13] S. Klaylat, Z. Osman, L. Hamandi, and R. Zantout, "Emotion recognition in arabic speech," *Analog Integrated Circuits and Signal Processing*, vol. 96, no. 2, pp. 337–351, 2018.

[14] M. Elmahdy, M. Hasegawa-Johnson, E. Mustafawi, R. Duwairi, and W. Minker, "Challenges and techniques for dialectal arabic speech recognition and machine translation," in *Qatar Foundation Annual Research Forum Volume 2011 Issue 1*, vol. 2011, no. 1. Hamad bin Khalifa University Press (HBKU Press), 2011, p. CSO5.

[15] A. Ali, S. Vogel, and S. Renals, "Speech recognition challenge in the wild: Arabic mgb-3," in *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2017, pp. 316–322.

[16] A. Ali, N. Dehak, P. Cardinal, S. Khurana, S. H. Yella, J. Glass, P. Bell, and S. Renals, "Automatic dialect detection in arabic broadcast speech," *arXiv preprint arXiv:1509.06928*, 2015.

[17] M. Akbacak, D. Vergyri, A. Stolcke, N. Scheffer, and A. Mandal, "Effective arabic dialect classification using diverse phonotactic models," in *Twelfth Annual Conference of the International Speech Communication Association*, 2011.

[18] R. Ekman, *What the face reveals: Basic and applied studies of spontaneous expression using the Facial Action Coding System (FACS)*. Oxford University Press, USA, 1997.

[19] A. H. Meftah, Y. A. Alotaibi, and S.-A. Selouani, "Evaluation of an arabic speech corpus of emotions: A perceptual and statistical analysis," *IEEE Access*, vol. 6, pp. 72 845–72 861, 2018.

[20] Y. Zhang and Q. Yang, "A survey on multi-task learning," *arXiv preprint arXiv:1707.08114*, 2017.

[21] ——, "An overview of multi-task learning," *National Science Review*, vol. 5, no. 1, pp. 30–43, 2018.

[22] Y. Shinohara, "Adversarial multi-task learning of deep neural networks for robust speech recognition." in *Interspeech*. San Francisco, CA, USA, 2016, pp. 2369–2372.

[23] S. Klaylat, L. Hamandi, R. Zantout, and Z. Osman, "Arabic natural audio dataset. mendeleydata, v1."

[24] R. Ziedan, M. Micheal, A. Alsammak, M. Mursi, and A. Elmaghraby, "A unified approach for arabic language dialect detection," in *29th International Conference on Computers Applications in Industry and Engineering (CAINE 2016), Denver, USA*, 2016.

[25] K. Almeman, M. Lee, and A. A. Almiman, "Multi dialect arabic speech parallel corpora," in *2013 1st International Conference on Communications, Signal Processing, and their Applications (ICCSPA)*. IEEE, 2013, pp. 1–6.

[26] M. Alsulaiman, G. Muhammad, M. A. Bencherif, A. Mahmood, and Z. Ali, "Ksu rich arabic speech database," *Information (Japan)*, vol. 16, no. 6 B, pp. 4231–4253, 2013.

[27] Y. Jia, Y. Zhang, R. Weiss, Q. Wang, J. Shen, F. Ren, P. Nguyen, R. Pang, I. L. Moreno, Y. Wu *et al.*, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," in *Advances in neural information processing systems*, 2018, pp. 4480–4490.

[28] Y. LeCun, L. Bottou, G. B. Orr, K.-R. Müller *et al.*, "Neural networks: Tricks of the trade," *Springer Lecture Notes in Computer Sciences*, vol. 1524, no. 5-50, p. 6, 1998.

[29] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, "Tensorflow: A system for large-scale machine learning," in *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, 2016, pp. 265–283.