Speaker Naming in Arabic TV Programs

Mohamed Lazhar Bellagha
Higher Institute of Computer Science and Communication Techniques ISITCom, University of Sousse, Tunisia
bellaghamohamed@gmail.com

Mounir Zrigui
Research Laboratory in Algebra, Numbers Theory and Intelligent Systems RLANTIS, University of Monastir, Tunisia
mounir.zrigui@fsm.rnu.tn

Abstract: *Automatic speaker identification is the problem of identifying speakers by their real identities. Previous approaches use textual information as a source of naming, try to associate names to neighbouring speaker segments using linguistic rules. However, these approaches have a few limitations that hinder their application on spoken text. Deep learning approaches for natural language processing have recently reached state-of-the-art results. However, deep learning requires a lot of annotated data which is difficult to obtain in the case of speaker identification task. In this paper, we present two contributions towards integrating deep learning for identifying speakers in news broadcasts: first we realise a dataset in which the names of mentioned speakers are related to the previous, next, current or other speaker turns. Moreover, we present our approach to solve the problem of speaker identification using information obtained from the transcription. We use a Long-term Recurrent Convolutional Network for name assignment and integer linear programming for name propagation into the different segments. We evaluate our model on both assignment and propagation tasks on the test part of the Arabic multi-genre broadcast dataset which consists of 17 TV programs from Aljazeera. The performance is analysed using the evaluation metrics, such as Estimated Global Error Rate (EGER) and Diarization Error Rate (DER). The outcome of the proposed method ensures better performance by achieving the lower EGER of 32.3% and DER of 8.3%.*

Keywords: *Speaker naming, speaker identification, name assignment, name propagation and CNN-LSTM.*

Received May 12, 2020; accepted May 27, 2021
https://doi.org/10.34028/iajit/19/6/1

1. Introduction

Person identification is an important and challenging task in multimedia research. It is the process of labelling each person with their real identity. Person identification has recently become an important key for many tasks, such as semantic indexing [16, 48], summarisation [39], or higher-level inference on multimedia data. It can be used for indexing TV programs and movies content.

Classical approaches on speaker identification use pre-trained biometric models, such as faces, and acoustic speaker models. The construction of these biometric models in a large collection of data is not useful due to the high variability of the content which makes these models not useful, and also due to their training time complexity [30]. A multimedia document may contain different information such as the speaker's identity, events and emotions. Hence, an alternative solution is to use information embedded in the document itself. Transcriptions coming from speech-recognition systems provide a potentially rich source of supervision [27, 42]. Moreover, TV program documents have a well-defined structure; the speech script is prepared in advance. For example, the presenter is responsible for presenting the people who will speak by their names using some clear and highly organised spoken sentences. Humans are able to understand who will speak only by listening to interlocutors and understanding contexts in which names are mentioned. These names are detected as person entities in the transcription [19, 20]. The best way to identify speakers by their full name is to use these named entities [11, 44]. The main objective of this work is to determine whether a detected entity refers to a speaker of the document or not.

Previous works tackle the issue of speaker identification by associating one of the four labels (current, previous, following, or other) to each name, using linguistic rules applicable to the context surrounding the mentioned name [22, 44]. These rules need to be programmed for every situation you want to distinguish, which needs to define a lot of linguistic rules for each situation. Furthermore, spoken language does not need to follow grammar theories and structures of the written language they belong to. This is what makes it difficult sometimes to define the meaning of a spoken sentence during a spoken speech, because the structure of the main components is not clear [43] such as the verb, the subject and the plural form, etc., For the Arabic language, it seems harder, which makes the use of linguistic rules more difficult [25]. Recently, deep learning and word embedding models have shown to be effective for natural language processing and have achieved good results in many tasks, such as sentiment analysis [2], plagiarism detection [26] and Author Identification [9], due to their ability to process sequences. These models can detect complicated patterns in written language. Moreover, deep learning methods have been successfully applied

for multimodal learning, where the visual, textual and acoustic data can be jointly processed in a supervised [6, 7, 37] or unsupervised manner [24]. Any created models for speaker naming need to know the linguistic structure of the sequence of words surrounding the mentioned name, in order to relate this name to their speaker turns or faces. However, there are some barriers to applying deep learning methods to name speakers using textual information (e.g., dialogue). One of the most important problems is the absence of annotated data (Speaker naming annotation). For this reason, the training of speaker naming architecture by relying on the transcription as the sole source of supervision would be particularly challenging, due to this lack of supervision between the textual and the audio domain. In each show, different speakers can contribute, only some of them are mentioned in the transcription, and possibly some mentioned names refer to a speaker that not contributing to the show. A more grounded form of annotation is therefore needed, to link the speaker intervention to the list of speakers who participated in the program, and to link the mention of a speaker's name in the transcription with the appearance of the corresponding speaker's voice. The existing datasets for speaker naming provide only two kinds of annotation: audio annotation with rich speech transcription and video annotation with head and embedded text annotation [4, 15]. The absence of speaker naming annotation prevented them from using deep learning approaches for naming speakers from the dialogue.

In this work, we present two contributions towards better integration of deep learning for speakers naming in TV programs: The first contribution consists in the construction of the dataset. We briefly present the Arabic Multi-Genre Broadcast (MGB) challenge, before describing the data used for speakers naming [3]. Then, we discuss the used methods to artificially create a specific corpus for speakers naming. Moreover, we will give some statistics about the capability of pronounced names to name speakers in the Arabic MGB-2 data. The second contribution relies on the proposed framework to assign names to speech turns. In the proposed framework we start by applying window technique to the textual sequence. Then, a trained Long-term Recurrent Convolutional Network (CNN-LSTM) model assigns one of the four labels (current, previous, another or following turn) to each name mentioned in this sequence. We use FARASA (named entity toolkit) [1] in order to extract pronounced names in transcription. Then we propose a clustering step based on integer linear programming in order to propagate these names to the different speaker turns. The assignment results are chosen as potential candidates in order to prevent the fusion of two speakers turns named differently. The proposed model achieves accuracy over 93% in the assignment step and an average identification error rate of about 32% in the final output on the test part of the MGB challenge dataset.

This paper is structured as follows: Section 2 discusses the previous works for speaker naming. The construction of a specific corpus for speaker naming is presented in section 3. Section 4 is briefly explained the proposed method for speaker naming. Consequently, section 5 provides the experimental results and performance analysis. Finally, section 6 concludes this paper.

2. Related Work

Our work is related to the use of natural language related to the multimedia content as a source of supervision. Natural language content can be in different forms such as transcriptions in TV programs or subtitles in movies. We present a brief overview of the related work in this area based on the type of modality used for extracting the names of speakers to name speakers' models.

- **Transcription**: The transcription of the audio signal obtained by an automatic speech recognition system (especially from television programs) is a rich source of information for identifying speakers [14, 28]. The basic assumption is that the speakers of the document are announced by their names during the show and that it is possible to use these names to identify the speakers.

Early works using speech transcripts as a source of identification are those proposed by Canseco Rodriguez *et al*., [10, 11]. They showed that the name of a speaker appearing in a given lexical context made it possible to identify the speakers of the document. They used manually defined language rules to assign a mentioned name to a speech segment. The idea is to use the linguistic context of each full name to assign one of the four labels: "previous speaker", "current speaker", "next speaker" or "other speaker" using some linguistic rules extracted from a corpus of television broadcasts. Following works, such as that of Tranter [44] proposed to automate the modelling of the lexical rules by the use of N-grams sequences learned from training data. Jousse *et al*. [22] have proposed an alternative to N-grams model by setting up an identification system based on a semantic classification tree. The proposed semantic tree allows to associate a probability to each occurrence of a full name, this probability indicates how this name corresponds to one of four classes (previous, current, next or other). They used diarization systems, speech recognition and automatically named entities. Petitrenaud *et al*. [31], propose to improve the semantic classification tree by a belief function.

The belief function takes account of contradictory information and uncertainty concerning the gender in order to manage conflict situations.

- **Written names**: when working with video data coming from TV broadcasts, there are two sources of names: names spoken in the audio stream and

names written on the screen in a specific text box to introduce people. The written names can be used as another source of information for speaker identification. Poignant *et al*. [33, 34] proposed an unsupervised speaker identification approach using the names written on the screen (obtained by an optical character recognition system). They proposed methods of naming that allow propagating the detected names towards the speaker turns, taking into account the duration of co-occurrence between the speaker turns and these names. [35] Adds a multimodal dimension to the identification task by integrating the knowledge provided by the written names to identify audio-visual clusters and to facilitate the process of diarization. The proposed system identifies both faces and speakers using only the names written on the screen as a source. Comparative works between the capabilities of written names and pronounced names in speaker identification show that automatic system extracts more hypothetical names from the speech, which means that pronounced names offer many instances of citations [5, 32].

- **Subtitles**: on the other hand, movies videos (films, TV series, etc.,) have the advantage of offering the script and subtitles as sources of additional information compared to TV programs (only subtitles are sometimes available). The subtitles contain the transcription of the dialogues (without the identity of the speaker) but are temporally aligned with the audio- visual source.

The script contains the transcription of the dialogues and other information related to the video (information about the speakers, scene,...), but no synchronisation information. The subtitles and scripts have been used as a source of supervision in order to improve speaker and face identification in movies. Haurilet *et al*. [17], propose naming characters in movies by analysing dialogue between the actors. This method is very important because it allows to use language description accompanying video to name characters. They used the names mentioned in the subtitles to obtain indications about the presence or absence of characters during a given period. To classify these names, the authors have defined 5 grammatical rules. These rules can classify the names into 3 categories in order to predict whether the bearer of this name will appear in a short time space or not. In the same context, Azab *el al*. [4] propose an unsupervised speaker-naming model in movies that use the names in subtitles extracted using the same grammatical rules in [17]. Then, they propose to combine textual, visual and acoustic modalities in a unified optimisation framework. Other works such as [36], identify speakers in movies using names extracted from textual resources through coreference resolution.

Pronounced names can be extracted from both transcription and subtitles. Naming people using these sources is challenging due to the difficulties of spoken language processing. Therefore, the use of linguistic rules has some limitations.

3. Proposed Corpus for Speaker Naming

The Arabic MGB Challenge [3] aims to support research on speech recognition, speaker diarization, dialect detection and alignment of audio broadcast, using TV recordings. A total of 1,200 hours have been crawled from Al-Jazeera channel over 10 years with lightly supervised transcriptions for the acoustic modelling. For language modelling, over 110M words are crawled from Aljazeera Arabic website.

The main goal of the challenge is to build a speech recognition system. The corpus contains about 3000 episodes which are divided into training, test and validation set. As described in [3], the recordings are accompanied by some metadata such as episode titles, program name, speakers' names and speakers' change information. Moreover, the recordings are split into different categories, which included multiple dialects.

Table 1 describes with more detail the recording contents. The main task in our work is to determine who speaks by their real identity. The goal is to combine acoustic and textual information coming from transcription. The speakers are cited in speech, for example, the presenter can mention the names of the interlocutors during the program. To answer this task, we artificially create a corpus for speakers naming in TV broadcast. The transcriptions of the Arabic MGB corpus are used to create the proposed corpus.

In this section, we give a statistical study about the capacity of pronounced names to name speakers. After that, we give details of our approach for building a speaker identification corpus.

Table 1. Overall statistics on the MGB data domain distribution.

Categories	- Conversation: 63% - Interview: 19% - Report: 18%
Domaine	- Politic: 76% - Economy: 8% - Society: 9% - Others: 7%
Duration	- All recordings have duration about 20-50 minutes
Spoken dialect	- Modern Standard Arabic: 70% - Dialect: 30% (multi-dialect)

3.1. Preliminary Study

We analyze the ability of pronounced names to name speakers on the training data of the MGB challenge. Over the training data, we detect 77,137 mentioned names, pronounced during the shows. Of this number, 34,760 names correspond to a person present (about 45% of pronounced names). Note that the presence of person refers only to the concerned show. If a person is mentioned in one show but their name is pronounced in another, it is not considered as present. The 34,760

pronounced names can name about 66, 7% of speakers in the training set. On the overall of the rest 33.3% non-named speakers, 56% are presenters. Some presenters try to introduce themselves, where this is not the case in most of the time. Hence, it is easy to create a specific biometric model to identify these categories of people. In this study, about 81% of mentioned names of present speakers are cited by presenters, when the presenter talks with more than one interlocutor.

3.2. Proposed Artificial Construction of Speaker Identification Corpus

Our approach for building a speaker identification corpus includes the following steps: First, we extract the names of person from transcription using FARASA named entity recognizer [1]. These names are compared to a pre-existing list which contains names of existing speakers. After that, we propose a method that allocates to each name one of the four labels: "next", previous", "current" or "other" speech turns. Finally, we apply a dynamic sliding window approach to each speech turn in order to capture the context surrounding the name.

- **Names extraction**: we use FARASA in order to detect names of people in transcriptions. We compare this list of names to a pre-existing list of names selected as candidates for being present in the show (using the Meta data provided by the MGB challenge). The proposed strategy takes into account the search of full name (first+last name) and partial name. In order to automatically determine that two names are similar, we used the edit distance to measure similarity between the data values. The edit distance between two strings is the minimum number of basic edit operations required to transform one string to the other. For the partial names we propose a supplementary treatment. We take only the names which are at the beginning or the end of sentences and preceded by nominal expressions (e.g., "doctor", "Mr"), based on the assumption that a characteristic of spoken language is that long dependent clauses are rare [17]. At the end of this process, we obtain a list of speech turns that contains the mentioned names of existing speakers.
- **Labeling speech turn**: to associate label to the speech turn we compare the mentioned names in speech turn by the name of the next, previous or current speaker (Figure 1). If any of these names does not correspond, we associate the label "other".
- **The dynamic sliding window**: speech turn may contain more than one name. In order to remove this ambiguity, a dynamic sliding window can be applied to the speech turn in order to extract the context surrounding the mentioned name, so that the speech turn contains only one name.

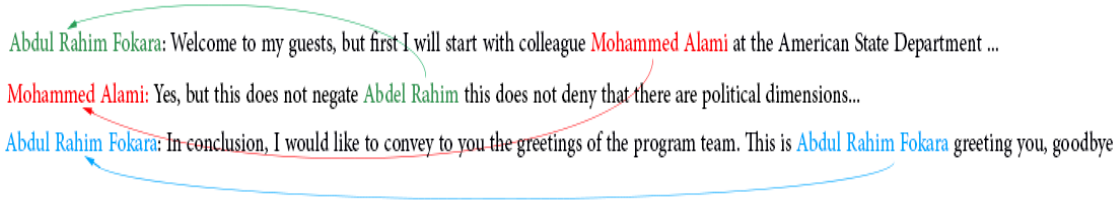

Figure 1. The basic principle of labeling speech turns, by tagging the mentioned names in transcript, for determining about whom the speaker is talking.

Table 2. Description of the collected speech turns.

Classes	Number of speech turns	Example
Next	10295	مرحبا بضيوفي لكن أبدأ بالزميل محمد العلمي في مقر وزارة الخارجية الأميركية Welcome to my guests, but first I will start with colleague Mohamed Alami at the American state department...
Previous	9023	شكراً للزميل محمد العلمي انضم إلي مشكوراً من مقر وزارة الخارجية الأميركية Thank you to colleague Mohammed Alami who joined me from the State Department
Current	626	في الختام أنقل لكم تحيات فريق البرنامج وهذا أحمد منصور يحييكم … In conclusion, I would like to convey the greetings of the program team. This is Ahmed Mansour greeting you ...
Other	9424	صحيح أنا لا أختلف مع الدكتور عبدالحميد على التوصيف لطبيعة هذه الثورات Certainly, I do not disagree with Mr. Abdul Hamid on the description of the nature of these revolutions...

At the end of this process, we collect a total of 29.441 speech turns, which are divided into four classes: The number of the speech turns that identify the next speaker's turn (10295), the number of the speech turns that identify the previous speaker's turn (9023), the number of the speech turns that identify the current speaker's turn (626), and the number of the speech turns that identify other speaker's turn (9424). Table 2 describes with more detail the collected data.

3.3. MGB-2 Test Data Annotation

We propose a manual annotation for the test part of the MGB-challenge. This annotation includes the speech turns and the acoustic speaker turns. The test part of the MGB challenge contains about 7 hours of Arabic TV-programs collected from Aljazeera Arabic website. Each of the records is about 25 min long with the text available for their transcription. To evaluate the speaker diarization performance, we propose to annotate 17

shows of the TV programs at the speaker level and the corresponding RTTM labels. The mentioned names during the show are manually labelled by one of the four labels "next", "previous", "current" or "other" at the sentence level. This data is used to demonstrate the name assignment approaches.

4. The Proposed Approach for Speaker Naming

In this section, we briefly describe the proposed model for speaker identification. As described in Figure 2, we start from audio recordings and their corresponding transcription. Over these inputs we apply feature extraction from audio and text. After that, we propose a CNN-LSTM text classification model to assign the appropriate labels to each speech turn. This step is called names assignment. It consists of assigning the name mentioned in the speech turns to the next, previous, current, or other speaker turns. Then, we propagate the assignment result into the different speaker segments using a clustering method based on Integer Linear Programming (ILP) and i-vectors speaker modeling. The assignment results are chosen as potential candidates in order to prevent the fusion of two speakers turns named differently during the propagation stage. Finally, we obtain segments annotated by the real identities of speakers.

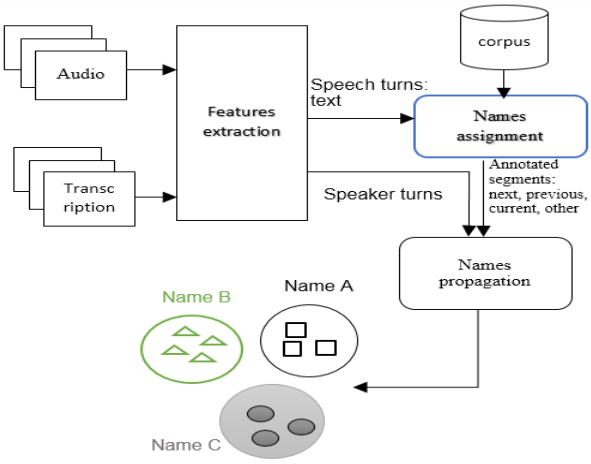

Figure 2. Summary of the proposed our approach for speaker identification using audio and its transcription.

Figure 2 presents A CNN-LSTM is trained to assign names to the next, previous, current or other speaker turns, while a clustering approach based on integer linear programming (ILP) is used for names propagation into speaker turns.

4.1. Feature Extraction

- **Textual features**: for the textual features, we use FARASA to detect names. Then, we use the dynamic sliding window approach described in the previous section to select the context of words surrounding the detected names.
- **Acoustic features**: we use 19 MFCC (Mel Frequency Cepstral Coefficient) and short time energy with their first and second order derivatives. We propose a manual segmentation for each recording and i-vectors approach to represent speaker segments by vectors. I-vector approach is the state-of-the-art in speaker recognition field. It consists in reducing a large-dimensional input data to a small-dimensional feature vector [12]. The i-vector algorithm is fully described in [8].

4.2. CNN-LSTM for Name Assignment

The model proposed for text classification is the multi-channel Convolutional Neural Network (CNN) [23] followed by Long-Short-Term Memory (LSTM) layer and two successive fully connected layers used as learnable classifiers (Figure 3). We use a pre-trained word embedding layer obtained from an unsupervised neural language model as input. Given a sequence of *n* words *W=W1, W2, …, Wn,* each word is represented by its their embedding vectors of dimension *D*. let *w* be a window of words w_i, w_{i+1}, …, w_{i+k}, the concatenated vector over this window is then:

$$\boldsymbol{w_{i:i+k} = w_i \oplus w_{i+1} \oplus \dots , \quad \oplus w_{i+k}} \qquad (1)$$

Over these sequences of vectors, the CNN uses a convolution operation with three filters *U=(u1, u2, u3)* applied to each possible window of N words in the sentences in order to produce a new feature map r_i:

$$\boldsymbol{r_i = g(w_{i:i+k}.U + b)} \qquad (2)$$

With, $b \in R$ being a bias term and g a non-linear function.

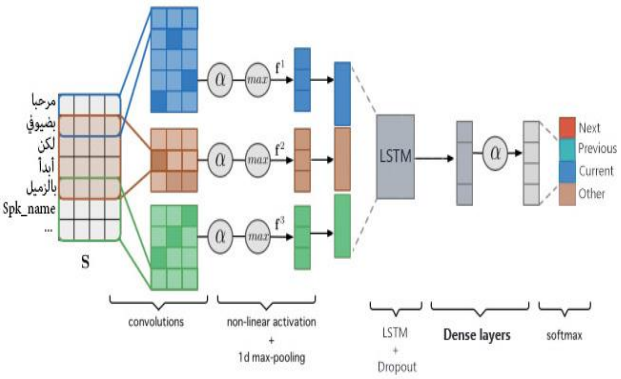

Figure 3. Model architecture with CNN and LSTM pipeline for name assignment.

As presented in Figure 3 the CNN composed of three filters. Every filter performs convolutions on the sentence matrix to generate feature maps. The 1D-max-over-time pooling is performed over each map. The generated features are concatenated to form a feature vector for the LSTM layer. The last two layers used as learnable layers to classify the sentence into four classes.

After that, we apply a max-over-time pooling operation which takes the maximum value of each features map. Max-over-time pooling means taking the maximum element of the feature map. This can capture the most important features [40, 46]. CNN can identify spatial patterns by using multiple filters with different sizes. These filters allow the sequences to be processed as different N-gram, which means that the model learns how to identify this interpretation regardless of their position. For sentence modelling, CNN perform excellently in extracting N-gram features at different positions of a sentence through the convolutional filters, and can learn short and long-range relations through pooling operations. It transforms each sentence into successive window (n-gram) features to help disentangle factors of variations within sentences. Interesting text analyses are based on the relationships between words, whether examining which words tend to follow others immediately, or that tend to co-occur within the same documents. So that, we give a unique and specific vector of embedding for each speaker name in the speech turns.

Then, over the concatenation of the different feature maps we use LSTM. LSTM [21] is a special kind of RNN specialized in learning long-term dependencies. LSTM unit updates the hidden state h at the time step t as follows:

$$\begin{aligned} i_t &= \sigma\left(x_t U^i + h_{t-1} W^i\right) \\ f_t &= \sigma\left(x_t U^f + h_{t-1} W^f\right) \\ o_t &= \sigma\left(x_t U^o + h_{t-1} W^o\right) \\ u_t &= tanh\left(x_t U^g + h_{t-1} W^g\right) \\ C_t &= \sigma(f_t * C_{t-1} + i_t * u_t) \\ h_t &= tangh\left(C_t\right) * o_t \end{aligned} \tag{3}$$

Where, i is the input gate, f is the forget gate, and o is the output gate. These gates control how the input is updated, how much the previous memory cell has forgotten, and the exposure of the memory, respectively at the time step t to form the hidden state vector.

Finally, the obtained features are passed to two Dense layers with respectively Relu and Softmax activation for outputting a prediction. For regularization, we use Dropout on the bottom layer of the CNN part with a constraint on l2-norms of the weight vectors [23]. Dropout is used also after the LSTM and the first Dense layers. Dropout is a regularization technique, for neural networks, that consists of randomly setting to zero a number of output features of the layer during backpropagation [18]. The CNN-LSTM can learn spatial and temporal patterns [13, 45], where CNN used to extract a sequence of higher-level phrase representations for the LSTM input to obtain the sentence representation. CNN-LSTM is able to capture both local features of phrases as well as global and temporal sentence semantics.

4.3. Name Propagation

The name propagation method is an ILP [38] clustering based. As described in [38], the goal of ILP clustering is to group N segments into K clusters, where the number of clusters K is determined by the algorithm. The objective function is to minimise the number of K classes, but also to minimise the dispersion of segments within each class.

$$Z = \sum_{n=1}^{N} yk + \frac{1}{F}\sum_{k=1}^{N}\sum_{n=1}^{N} d(w_k, w_n)x_{k,n} \tag{4}$$

Where $\sum_{n=1}^{N} yk$ calculates the number of classes in the problem and $\frac{1}{F}\sum_{k=1}^{N}\sum_{n=1}^{N} d(w_k, w_n)x_{k,n}$ calculates the sum of the distances between the centre of the class k and the segments attached to this class, or ($d(w_k,w_n)$ is the distance between the centre of the class k and the segment n.

Thus, the clustering model can be written as:

$$\begin{aligned} &Minimize: \quad Z \\ &Subject\ To: \quad \sum_{n=1}^{N} x_{k,n} = 1, \quad \forall k, \end{aligned} \tag{5}$$

$$x_{k,n} - y_k \leq 0, \quad \forall k, \forall n \tag{6}$$

$$d(w_k, w_n)x_{k,n} \leq \delta, \quad \forall k, \forall n \tag{7}$$

$$x_{k,n} \in \{0,1\}, \quad \forall k, \forall n \tag{8}$$

$$yk \in \{0,1\}, \quad \forall k \tag{9}$$

Where, Equation (5) ensures that each i-vector n must be associated to only one center k. Equation (6) ensures that if an i-vector n is assigned to a class k, then the class k is selected. In Equation (7), the i-vector n associated with the center k (i.e., $x_{k,n}$=1) must have a distance $d(w_k,w_n)x_{k,n}$ less than a threshold δ determined experimentally. In order to prevent the fusion of two speakersâ€™ turns named differently we propose to maximize the distance between these segments before clustering. We set the distance between the segments named differently as maximal distance. The Mahalanobis distance is used to calculate the acoustic similarity and create the matrix distance. Finally, over this matrix we apply ILP clustering to fuse the similar i-vectors.

5. Experimental setup and Evaluation

As described in the previous section, the proposed architecture is made up of two main modules: the first one is the assignment process, which consists in classifying speech turns into four classes. While the second module is for name propagation into acoustic speaker turns. We evaluate the performance of different neural network models with respect to the name assignment module, and we evaluate two different fusion strategies for name propagation. We train the neural model on the proposed corpus which consists of 29,441 speech turns split into 75% for training and 25%

for validation. Moreover, we use the test part of the MGB-challenge to evaluate both the quality of the assignment and propagation strategies. The data used on the test part are segmented manually, and each segment is represented using i-vectors. We used 60-dimensional acoustic features, composed of 19 MFCCs plus log energy and augmented by first and second-order derivatives. We have chosen a dimension of 75 for the i-vectors. We implement our models based on tensorflow: a python library, which supports efficient symbolic differentiation and transparent use of a GPU. To benefit from the efficiency of parallel computation of the tensors, we train the models on a GPU.

5.1. Pre-trained Word Vectors

We use a pre-trained word embedding[1] obtained from an unsupervised neural language model trained by word2vec [29], and the word vectors have dimensionality of 300 and were trained using the continuous Skip-gram architecture. To accomplish our work, we collect a large Arabic broadcast news corpus from different resources. In addition to the classical pre-processing techniques used for cleaning corpora, we propose to link the named entities (PERSON, LOCATION and ORGANIZATION) with "_" to be considered as single word. For example, (المملكة العربية السعودية, Kingdom of Saudi Arabia) becomes (المملكة_العربية_السعودية, Kingdom_of_Saudi_Arabia).

5.2. Evaluation Metrics

First, we introduce the measures used to evaluate the classification model for name assignment. We use the classical measures used in information retrieval the precision, recall, Accuracy and F-measure. In addition, We use the diarization error rate (DER) to evaluate clustering results. The DER is defined as:

$$DER = \frac{\textbf{false alarm} + \textbf{missed detection} + \textbf{confusion}}{\textbf{Total}} \quad (10)$$

Where false alarm is the error duration of false alarm, missed detection is the error duration of missed detection, confusion is the duration of speaker confusion, and total is the total duration of speech.

The next metric is the Estimated Global Error Rate (EGER) to measure the quality of the identification results (final output). The EGER metric is defined as:

$$EGER = \frac{\text{false alarm} + \text{missed detection} + \text{confusion}}{\#Total} \quad (11)$$

Which is similar to the Diarization Error Rate (DER) introduced previously, except that the confusion term is computed directly by comparing reference and hypothesis labels, and does not rely on a prior one-to-one matching.

[1]https://github.com/MohamedBellagha/Arabic-Word-Embedding

5.3. Evaluation Results on Name Assignment

We define three architectures of neural network: the first neural network model is a staked of bidirectional Long-short-term memory (bi-LSTM). It is composed of an embedding layer and three bi-LSTM layers with 256, 128, 128 units respectively. The second neural model is the Kim Yon CNN architecture with multiple filters. We use multiple sizes of filters (3 filters of size 3, 5 et 7). The last model is the proposed one. As described in the previous section, it is a combination of CNN and LSTM. For the filter size, we investigated filter lengths of 3, 5 and 7 in multiple convolutional layers with different lengths of filters in parallel. For the first case, each n-gram window is transformed into n_i convoluted features after convolution (where n_i denote the number of filters of length i) and the sequence of window representations is fed into LSTM.

For the latter case, since the number of windows generated from each convolution layer varies when the filter length varies, we cut the window sequence at the end based on the maximum filter length that gives the shortest number of windows.

Table 3. Accuracy, F1-score, Recall and precision of names assignment on the test set of the MGB challenge by comparing the proposed CNN-LSTM with SVM, Logistic Regression (LR), BI-LSTM and the multichannel CNN.

Model	Results	Reported in
SVM	Accuracy: 86.3%	Socher *et al*. (2013) [41]
LR	Accuracy: 85.18%	Our implementation
CNN	Accuracy: 91.89% Recall: 91.89% Precision: 91.95%	Kim (2014) [23]
Bi-LSTM	Accuracy: 91.63% Recall: 91.63% Precision: 91.63%	Our implementation
CNN-LSTM	Accuracy: 93.32% Recall: 93.32% Precision: 93.36%	Our implementation

Each window is represented as the concatenation of outputs from different convolutional layers. We also exploit different combinations of different filter lengths. In the convolutional layer of our model, filters are used to capture local n-gram features. According to [47], multiple convolutional layers in parallel with different filter sizes should perform better than single convolutional layers with the same length filters in that different filter sizes could exploit features of different n-grams.

The learnable classifiers are two fully connected layers with 50 and 4 units with Relu and sigmoid activation, respectively, used for the three models. We use a set of hyper-parameters during the training. We set the batch size at 64 and we reduce the learning rate by a factor of 2-10 once learning stagnates when the accuracy metric has stopped improving through the Adam optimizer. We use the accuracy on the validation set to locate the best epoch and best hyper-parameter

settings for testing. For other baseline methods, we compare SVM with average word vector features and LR with average word vector features. The main difference is that features used come from averages of word embeddings, specifically word2vec vectors. It consists of transforming the resulting vector points to a single vector summarising the whole sentence. As reported in Table 3, We show the results in terms of accuracy by comparing the proposed CNN-LSTM model described in section 4.1, with bidirectional LSTM and the multichannel CNN [23]. As can be seen, the proposed CNN-LSTM model obtains the best results on the test sets.

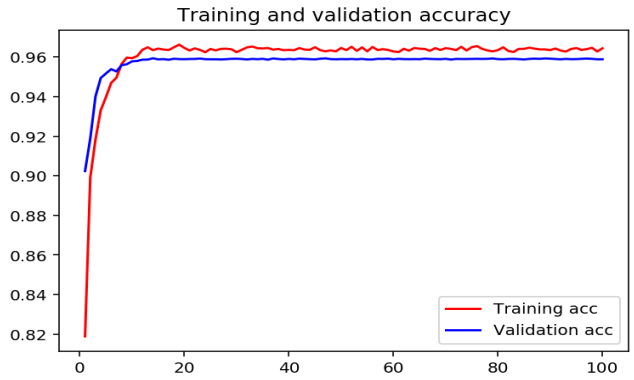

Figure 4. Training and validation accuracy for names assignment when using CNN-LSTM on the created corpus.

The CNN-LSTM mixed model precision rate is 7.02% and 8.14% higher than the traditional machine learning SVM and LR methods, respectively, and 1.43% and 1.69% higher than the traditional neural network CNN and BI-LSTM methods, respectively. Therefore, the precision of the proposed hybrid model is higher than that of the other methods, this is because the hybrid model not only captures CNN's ability to extract local features, but also combines LSTM to preserve historical information and extract contextual dependencies of text, at the same time, the deep learning method is used to avoid the manual feature extraction and dimensional reduction operation, and the classification effect is better. On the other hand, the performance obtained by the CNN-LSTM during the training phase is reported in Figure 4. Looking at the training and validation curves above, it seems that the model's training is going well and the accuracy is approaching 97%.

From the obtained results, we have the following observations:

- Our result consistently outperforms all classical machine learning models and neural baseline models, which means that CNN-LSTM captures intentions of name classification.
- Our result outperforms SVM and LR that depends on highly engineered features. With the ability of automatically learning semantic sentence representations, CNN-LSTM doesn't require any designed features and has a better scalibility.
- From the confusion matrix illustrated in Figure 5, which exhibits for the CNN-LSTM, the higher accuracy (94%) for the class 'other', but a moderate accuracy (82%) for the class 'current', while the other classes obtain better performance.

5.4. Evaluation Results on Name Propagation

We evaluate two strategies for name propagation. The first strategies prevent the fusion of two speakers' turns named differently in the assignment step. The second method uses an optimal mapping based on the Hungarian algorithm to compute the matching between the diarization output and the assignment output. Moreover, we report the results of the final output, and we compare these results with an oracle which corresponds to the best possible performance. The test corpus is composed of 17 different types of shows divided into four different programs.

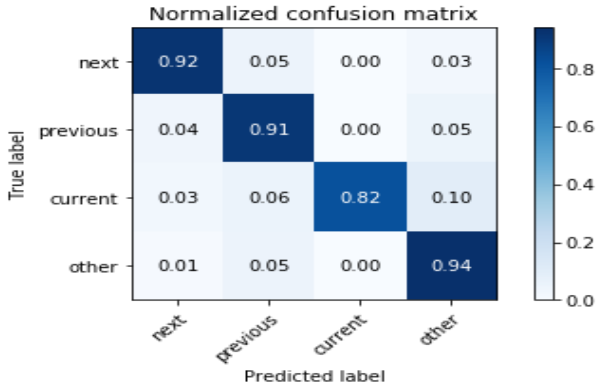

Figure 5. Normalized confusion matrix of the CNN-LSTM model on the test subset.

Table 4. Diarization error rate and identification error rate on the test part of the MGB challenge, using two fusion strategies.

Metric\System	Prevent fusion		Optimal mapping	
	EGER	DER	EGER	DER
SVM	35.8%	12.2%	34.9%	11.1%
LR	36.72%	12.8%	35.1%	11.1%
CNN	33.8%	9.2%	34.6%	11.1%
Bi-LSTM	33.5%	8.9%	34.6%	11.1%
CNN-LSTM	**32.3%**	8.3%	33.8%	11.1%

The obtained results (Table 4) show that the CNN-LSTM approach for speaker naming yields the best results (EGER=32.3% and DER=8.3%) compared to the results given by the other classifiers. We also show that the assignment results can improve the diarization results when preventing the fusion of some speech turns, but it is conditioned by the results given by the classifiers. In our case, this improvement on the DER appears only when using CNN, Bi-LSTM, and CNN-LSTM. In the evaluation results, we remark that the majority of non-named speech turns are from presenters. The presenter tries to introduce the interlocutors during the show, however, there are a few of them who introduce themselves.

- **Compared with Previous Works**. Previous works

[31, 35] have addressed similar Speaker naming problems by incorporating speaker turns, face, embedded text in screen and speech turns. They evaluated on REPERE and achieved speaker naming EGER of 14.9% and 20.1% respectively. In comparison, we can achieve speaker naming EGER of 32.3% on our dataset without introducing any face/person tracking, or embedded text on the screen.

6. Conclusions

This paper presents two contributions to speaker identification in Arabic news broadcast. The first contribution is related to the proposed dataset for speaker identification. The dataset is an expansion of the Arabic MGB corpus in which each name appearance in speech turns is located and annotated according to its appropriate speaker turn. The second contribution is related to the proposed approach to name each speech turn. The proposed approach combines CNN-LSTM for labeling each detected name and I-vector+ILP for names propagation. For name propagation, we propose two methods. The first one prevents the fusion of two speakersâ€™ turns named differently in the assignment step. The second method uses the optimal mapping between the diarization and the assignment output. The results on the test phase of our corpus show that the CNN-LSTM obtained an accuracy of 93,3% and an EGER of 32.3% in the final output. In this work we propose using a pre-existing list of names, which is not always available for all programs. Hence, as future work, we will try to automatically extract the list of existing people from transcription. In this case deep learning can be used to jointly tackle the problem of the presence of people and name assignment. In the preliminary study of the naming capabilities of names pronounced in the Arabic MGB data, we show that 56% of non-named people are presenters. Hence, we conceive that it is useful to use some biometric models for a particular person.

References

[1] Abdelali A., Darwish K., Durrani N., and Mubarak H., "Farasa: A fast and furious segmenter for Arabic," *in Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, San Diego, pp. 11-16, 2016.

[2] Abdellaoui H. and Zrigui M., "Using Tweets and Emojis to Build TEAD: an Arabic Dataset for Sentiment Analysis," *Computación y Sistemas*, vol. 22, no. 3, pp. 777-786, 2018.

[3] Ali A., Bell P., Glass J., Messaoui Y., Mubarak H., Renals S., and Zhang Y., "The MGB-2 Challenge: Arabic Multi-Dialect Broadcast Media Recognition," *in Proceedings of the Spoken Language Technology Workshop*, San Diego, pp. 279-284, 2016.

[4] Azab M., Wang M., Smith M., Kojima N., Deng J., and Mihalcea R., "Speaker Naming in Movies," *in Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, New Orleans, pp. 2206-2216, 2018.

[5] Bechet F., Favre B., and Damnati G., "Detecting Person Presence in TV Shows with Linguistic and Structural Features," *in Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, Kyoto, pp. 5077-5080, 2012.

[6] Bellagha M. and Zrigui M., "Speaker Naming in TV Programs Based on Speaker Role Recognition," *in Proceedings of IEEE/ACS 17th International Conference on Computer Systems and Applications*, Antalya, pp. 1-8, 2020.

[7] Bellagha M. and Zrigui M., "Using the MGB-2 Challenge Data for Creating a New Multimodal Dataset for Speaker Role Recognition in Arabic TV Broadcasts," *Procedia Computer Science*, vol. 192, pp. 59-68, 2021.

[8] Bousquet P., Matrouf D., and Bonastre J., "Intersession Compensation and Scoring Methods in the I-Vectors Space for Speaker Recognition," *in Proceedings of 12th Annual Conference of the International Speech Communication Association*, Florence, 2011.

[9] Bsir B. and Zrigui M., "Bidirectional LSTM for Author Gender Identification," *in Proceedings of International Conference on Computational Collective Intelligence*, Bristol, pp. 393-402, 2018.

[10] Canseco-Rodriguez L., Lamel L., and Gauvain J., "A Comparative Study Using Manual and Automatic Transcriptions for Diarization," *in Proceedings of Automatic Speech Recognition and Understanding, IEEE Workshop*, Cancun, pp. 415-419, 2005.

[11] Canseco-Rodriguez L., Lamel L., and Gauvain J., "Speaker Diarization from Speech Transcripts," *in Proceedings of the 8th International Conference on Spoken Language Processing ICC*, Jeju, 2004.

[12] Dehak N., Kenny P., Dehak R., Dumouchel P., and Ouellet P., "Front-End Factor Analysis for Speaker Verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788-798, 2011.

[13] Donahue, J., Anne Hendricks, L., Guadarrama S., Rohrbach M., Venugopalan S., Saenko K., and Darrell T., "Long-Term Recurrent Convolutional Networks for Visual Recognition and Description," *in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, pp. 2625-2634, 2015.

[14] Esteve Y., Meignier S., Deléglise P., and

Mauclair J., "Extracting true speaker identities from transcriptions," *in Proceedings of the 8th Annual Conference of the International Speech Communication Association*, Antwerp, pp. 2601-2604, 2007.

[15] Giraudel A., Carré M., Mapelli V., Kahn J., Galibert O. and Quintard L., "The REPERE Corpus: a Multimodal Corpus for Person Recognition," *in Proceedings of the 8th International Conference on Language Resources and Evaluation*, Istanbul, pp. 1102-1107, 2012.

[16] Haffar N., Hkiri E., and Zrigui M., "Using Bidirectional LSTM and Shortest Dependency Path for Classifying Arabic Temporal Relations," *Procedia Computer Science*, vol. 176, pp. 370-379, 2020.

[17] Haurilet M., Tapaswi M., Al-Halah Z., and Stiefelhagen R., "Naming TV Characters by Watching and Analyzing Dialogs," *in Proceedings of the Applications of Computer Vision, IEEE Winter Conference*, Lake Placid, pp. 1-9, 2016.

[18] Hinton G., Srivastava N., Krizhevsky A., Sutskever I., and Salakhutdinov R., "Improving Neural Networks by Preventing Co-Adaptation of Feature Detectors," *arXiv preprint arXiv:1207.0580*, 2012.

[19] Hkiri E., Mallat S., and Zrigui M., "Events Automatic Extraction from Arabic Texts," *International Journal of Information Retrieval Research*, vol. 6, no. 1, pp. 36-51, 2016.

[20] Hkiri E., Mallat S., Zrigui M., and Mars M., "Constructing a Lexicon of Arabic-English Named Entity using SMT and Semantic Linked Data," *The International Arab Journal of Information Technology*, vol. 14, no. 6, pp. 820-825, 2017.

[21] Hochreiter S. and Schmidhuber J., "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997.

[22] Jousse V., Meignier S., Jacquin C., Petitrenaud S., Estève Y., and Daille B., "Analyse Conjointe Du Signal Sonore Et De Sa Transcription Pour L'identification Nommée De Locuteurs," *Traitement Automatique Des Langues*, vol. 50, no. 1, pp. 201-225, 2009.

[23] Kim Y., "Convolutional Neural Networks for Sentence Classification," *arXiv preprint arXiv:1408.5882*, 2014.

[24] Le N. and Odobez J., "Learning Multimodal Temporal Representation for Dubbing Detection In Broadcast Media," *in Proceedings of the 24th ACM international Conference on Multimedia*, Amsterdam, pp. 202-206, 2016.

[25] Lhioui C., Zouaghi A. and Zrigui M., "Towards a Hybrid Approach to Semantic Analysis of Spontaneous Arabic Speech," *International Journal of Computational Linguistics and Applications*, vol. 5, no. 2, pp. 165-193, 2014.

[26] Mahmoud A. and Zrigui M., "Semantic Similarity Analysis for Corpus Development and Paraphrase Detection in Arabic," *The International Arab Journal of Information Technology*, vol. 18, no. 1, pp. 1-7, 2021.

[27] Maraoui M., Terbeh N., and Zrigui M., "Arabic Discourse Analysis Based on Acoustic, Prosodic and Phonetic Modeling: Elocution Evaluation, Speech Classification and Pathological Speech Correction," *International Journal of Speech Technology*, vol. 21, no. 4, pp. 1071-1090, 2018.

[28] Mauclair J., Meignier S. and Esteve Y., "Speaker diarization: About Whom The Speaker Is Talking?," *in Proceedings of the IEEE Odyssey-The Speaker and Language Recognition Workshop*, San Juan, pp. 1-6, 2006.

[29] Mikolov T., Sutskever I., Chen K., Corrado G., and Dean J., "Distributed Representations of Words and Phrases and Their Compositionality," *in Proceedings of the 26th International Conference on Neural Information Processing Systems*, Tahoe Nevada, pp. 3111-3119, 2013.

[30] Moattar M. and Homayounpour M., "A Review on Speaker Diarization Systems and Approaches," *Speech Communication*, vol. 54, no. 10, pp. 1065-1103, 2012.

[31] Petitrenaud S., Jousse V., Meignier S. and Estève Y., "Identification of speakers by name using belief functions," *in Proceedings of the International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, Dortmund, pp. 179-188, 2010.

[32] Poignant J., Besacier L., and Quénot G., "Nommage Non-Supervisé Des Personnes Dans Les Emissions De Télévision: Une Revue Du Potentiel De Chaque Modalité," *CORIA 2013, Papier Long (oral)*, 2013.

[33] Poignant J., Bredin H., Besacier L., Quénot G., and Barras C., "Towards a better Integration of Written Names for Unsupervised Speakers Identification In Videos," *in Proceedings of the 1st Workshop on Speech, Language and Audio in Multimedia, SLAM*, Marseille, pp. 84-89, 2013.

[34] Poignant J., Bredin H., Le V., Besacier, L., Barras C. and Quénot G., "Unsupervised Speaker Identification Using Overlaid Texts in TV Broadcast," *in Proceedings of the Interspeech Conference of the International Speech Communication Association*, Portland, pp. 2650-2653, 2012.

[35] Poignant J., Fortier G., Besacier L., and Quénot, G., "Naming Multi-Modal Clusters to Identify Persons in TV broadcast," *Multimedia Tools and Applications*, vol. 75, no. 15, pp. 8999-9023, 2016.

[36] Ramanathan V., Joulin A., Liang P., and Fei-Fei L., "Linking People in Videos with "Their" Names Using Coreference Resolution,"

Proceedings of the European Conference on Computer Vision, Zurich, pp. 95-110, 2014.
[37] Ren J., HuY., Tai Y., Wang C., Xu L., Sun W., and Yan Q., "Look, Listen and Learn-A Multimodal LSTM for Speaker Identification," *in Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, Phoenix Arizona, pp. 3581-3587, 2016.
[38] Rouvier M. and Meignier S., "A Global Optimization Framework for Speaker Diarization," *Odyssey*, 2012.
[39] Sghaier M. and Zrigui, M., "Rule-Based Machine Translation from Tunisian Dialect to Modern Standard Arabic," *KES 2020*, Verona, 2020.
[40] Shen D., Min M., Li Y., and Carin L., "Adaptive Convolutional Filter Generation for Natural Language Understanding," *CoRR*. abs/1709.08294, 2017.
[41] Socher R., Perelygin A., Wu J., Chuang J., Manning C., Ng A., and Potts C., "Recursive Deep Models for Semantic Compositionality Over A Sentiment Treebank," *in Proceedings of the Conference on Empirical Methods In Natural Language Processing*, Seattle, pp. 1631-1642, 2013.
[42] Terbeh N. and Zrigui M., "Vers La Correction Automatique De La Parole Arabe," *Citala 2014*, 2014.
[43] Terbeh N. and Zrigui M., "Vocal Pathologies Detection and Mispronounced Phonemes Identification: Case of Arabic Continuous Speech," *in Proceedings of the 10th International Conference on Language Resources and Evaluation*, Portorož, pp. 2108-2113, 2016.
[44] Tranter S., "Who Really Spoke When? Finding Speaker Turns and Identities in Broadcast News Audio," *in Proceedings of the Acoustics, Speech and Signal Processing, ICASSP IEEE International Conference*, Toulouse, 2006.
[45] Vinyals O., Toshev A., Bengio S., and Erhan D., "Show and Tell: A Neural Image Caption Generator," *in Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, Boston, pp. 3156-3164, 2015.
[46] Zhang Y. and Wallace B., "A Sensitivity Analysis of (and Practitioners' Guide To) Convolutional Neural Networks for Sentence Classification," *arXiv preprint arXiv:1510*, 2015.
[47] Zhou C., Sun C., Liu Z., and Lau F., "A C-LSTM Neural Network for Text Classification," *arXiv preprint arXiv:1511.08630*, 2015.
[48] Zrigui M., Charhad M., and Zouaghi A., "A Framework of Indexation and Document Video Retrieval Based on the Conceptual Graphs," *Journal of Computing and Information Technology*, vol. 18, no. 3, pp. 245-256, 2010.

Mohamed Lazhar Bellagha a PhD student in the Higher Institute of Computer Science and Communication Techniques ISITCom, Hammam Sousse, Tunisia. He is a member of Research Laboratory in Algebra, Numbers Theory and Intelligent Systems RLANTIS, Monastir, Tunisia. His areas of interest include Speaker identification, machine learning and natural Language Processing.

Mounir Zrigui a full professor at the University of Monastir, Tunisia. He received his PhD from the Paul Sabatier University, Toulouse, France in 1987 and his HDR from the Stendhal University, Grenoble, France in 2008. Since 1986, he is a Computer Science Assistant Professor in Brest University, France, and after in the Faculty of Science of Monastir, Tunisia. He has started his research, focused on all aspects of automatic natural language processing (written and oral). He has run many research projects and published many research papers in reputed international journals/ conferences.