Automatic Recognition of Arabic Poetry Meter from Speech Signal using Long Short-term Memory and Support Vector Machine

Abdulbasit K. Al-Talabani

Department of Software Engineering, Faculty of Engineering, Koya University,

Koya KOY45, Kurdistan Region - F.R. Iraq

***Abstract*—The recognition of the poetry meter in spoken lines is a natural language processing application that aims to identify a stressed and unstressed syllabic pattern in a line of a poem. State-of-the-art studies include few works on the automatic recognition of Arud meters, all of which are text-based models, and none is voice based. Poetry meter recognition is not easy for an ordinary reader, it is very difficult for the listener and it is usually performed manually by experts. This paper proposes a model to detect the poetry meter from a single spoken line ("Bayt") of an Arabic poem. Data of 230 samples collected from 10 poems of Arabic poetry, including three meters read by two speakers, are used in this work. The work adopts the extraction of linear prediction cepstrum coefficient and Mel frequency cepstral coefficient (MFCC) features, as a time series input to the proposed long short-term memory (LSTM) classifier, in addition to a global feature set that is computed using some statistics of the features across all of the frames to feed the support vector machine (SVM) classifier. The results show that the SVM model achieves the highest accuracy in the speaker-dependent approach. It improves results by 3%, as compared to the state-of-the-art studies, whereas for the speaker-independent approach, the MFCC feature using LSTM exceeds the other proposed models.**

***Index Terms*—Speech processing, Long short-term memory, Support vector machine, Prosody, Cepstral features.**

I. Introduction

Arabic poetry prosody (APP) ("Arud" عروض in Arabic) is the science that measures classical Arabic rhymed poetry (Stoetzer, 1989). Poetry meters ("Bahr" بحر or "Wazn وزن" in Arabic) are a sequence of diacritics representing a stressed and unstressed syllabic pattern adopted in APP to classify poetries into different classes based on special rules called meters. APP has been adapted to other eastern poetries such as Kurdish, Persian, and Urdu poetries (Kurta and Kara, 2012). A poem's meter is important for evaluating the commitment of classical poems to the musical flow. However, unlike understanding a poem's meaning, detecting a poem's meter is difficult for the ordinary reader, is much harder for listeners, and is mostly done by experts.

Arud meter recognition (AMR) in Arabic poetry is a much-studied topic. Most of these studies are theoretical and few are practical. The history of theoretical studies in Arud metrics is very old and dates back to the founder of the science, Al-Khalil bin Ahmad Al-Farahidi (AD 718-786) (Arberry, 1965), and addresses the theoretical problems of this science (Morris, 1966). However, few works have been done regarding automatic AMR (AAMR), and none have adopted AAMR using speech signals, but rather they are text-based proposed models. For example, Ismail et al. (2010) developed a prototype, expert system harmony test to provide expert-level solutions for testing harmony correction and to identify the pattern ("Bahr") of Arabic poetry. In Alnagdawi et al. (2013), the authors implemented a tool to find an Arabic poem meter using context-free grammar. They used trimmed poems (words with "Tashkeel") to detect the meter in Arabic poetry. Abuata and Al-Omari (2018) introduced an algorithm that can determine the correct meter for a given Arabic poem and are also able to convert the poem into Arud Writing. The algorithm is based on a set of well-defined rules applied only to the first part ("Sadr") of the poem's verse. In more advanced work, Yousef et al. (2019) have recognized 16 meters of Arabic poetry and four meters of English from the plain text using recurrent neural network (RNN) with an overall accuracy of 96.38% and 82.31%, respectively.

Another work close to the automatic recognition of Arud meters (AARM) model that is found in the literature is rhyme ("Qafiah قافيه") detection. For instance, Hirjee and Brown (2010) developed a method to score potential rhymes using a probabilistic model based on phoneme frequencies in rap lyrics. However, the work adopted the use of lyric text as an input rather than vocal input. We are not aware of any academic work that takes vocal or audio rendition poetry as an input to the AARM model.

AARM, using audio, aims to analyze the speech signal and uses its features to detect meters from the read poem. This work's main contribution is to introduce a model to recognize the meters from a voice recording of a spoken line (Bayt) of

ARO-The Scientific Journal of Koya University
Vol. VIII, No.1(2020), Article ID: ARO.10631, 5 pages
DOI: http://dx.doi.org/10.14500/aro.10631
Received 03 February 2020; Accepted: 27 March 2020
Regular research paper: Published 14 April 2020
Corresponding author's e-mail: abdulbasit.faeq@koyauniversity.org

an Arabic poem. The application proposed in this paper will help learners recognize the meters of a poem through reading instead of writing it down. It could be also adapted to help learners to improve their pronunciation of the classical poem in the correct way, as a computer-based assessment of language speaking skills (Araújo, 2010). The poetry meter has also been used as a feature to detect the authorship of unknown poems (Al-Falahi, et al., 2017). Poetry meters could also be used to improve studies that try to find out poets' differences and similarities through analyzing their works using natural language understanding technologies (Zhang and Gao, 2017).

The data used in this work are 230 single poem lines recorded by two male readers of three different meters (Taweel, Kameel, and Baseet), which accounted as the most prevalent meters, and where Taweel is the most used due to its simplicity (Scott, 2010). This work aims to detect the poetry meter, based on the speech prosody characteristics of the recording signal. We proposed to extract the linear prediction cepstrum coefficient (LPCC) and Mel frequency cepstral coefficient (MFCC) cepstral features, where the cepstral coefficients are derived from either LP analysis or a filter bank approach and are almost treated as standard front end features (Rao and Koolagudi, 2013). Cepstral features have achieved a high level of accuracy for speech recorded in a non-noisy environment (Reynolds, 1995) and are used as an essential feature in speaker verification (Sarangi and Saha, 2020). The poetry meter data should have a time series nature because it depends on how the patterns ("Tafa'il تفاعيل") come next to each other. The classifier with a time series nature is expected to be more suitable for such an application. In this paper, we adopt the use of long short-term memory (LSTM) (Hochreiter and Schmidhuber, 1997). LSTM is a form of RNN and has been proposed to overcome the problem of vanishing gradient from which RNN suffers. The model changes the standard RNN hidden layer, where each ordinary node in the hidden layer is replaced by a memory cell (Lipton, et al., 2015). In this work, the classifier takes the LPCC and MFCC features and maps each input into a single label or Arud meter. However, we can also deal with the data as a non-time series signal features where the order of the patterns' features is not important, but rather are the global feature of all of the lines of read poetry. Statistics such as mean and standard deviation is used in this work to globalize the features to later feed the support vector machine (SVM) classifier (Vapnik, 1995).

The rest of this paper is organized as follows: In Section II, an entry to the Arabic prosody science (Arud) is presented, followed by the scope of this work in Section III. Section IV presents the methodology needed for this work. Finally, a discussion of the results and a conclusion are reported in Sections V and VI, respectively.

II. Arabic Poetry Prosody (APP)

Classical Arabic poems have features common to poems written in some other eastern languages, whereas some features are unique to Arabic poetry. The science was established in the pre-Islamic era and has remained almost unchanged until now (Almuhareb, et al., 2013). The main two types of Arabic poetry are, rhymed or measured and prose. Al-Khalīl bin Ahmad Al-Farahidi is the founder of rhymed, which is called Arud in Arabic. Arud is the science that studies the meter used in classical Arabic and some other eastern languages. Al-Farahidi wrote 15 meters ("seas," Bahr in Arabic) and was followed by another meter written by his student Al-Akhfash giving a total of 16 meters (Yousef, et al., 2019). The measuring unit of meters is called Patterns ("Tafa'il"), in which each contains a certain number of Tafā'l, which the poet follows in every line (Bayt) of the poem. The original main patterns are shown in Table I Abuata and Al-Omari (2018). Classic Arabic poetry ends with the same rhyme (qafiyah), and each meter has its own key which is followed in each line along with the poem. The key consists of a list of patterns that represent the order of the consonant and vowel sounds presented in each Bayt. The pattern could also be represented in terms of scansion (Movement "Harakah" or Stillness "Sukun"). Table II presents the three meters included in this study with their scansion (Yousef, et al., 2019).

III. The Scope of the Work

In this paper, we claim that a poem's meters can be automatically recognized by a spoken line of poetry. In addition, this work will investigate the hypothesis of how a time series representation of the spoken poetry line is different from a non-time series representation. Various meters are supposed to have a different sequence of patterns based on each meter (Figs. 1 and 2). Consequently, whenever the poetry line is read correctly the sequence of features of each speech signal will reflect the sequence of the patterns. This work aims to design a model to map the stream of feature values or a global feature for each poetry line into the correct meter that the poem follows.

IV. Methodology

As in any pattern recognition process, the feature extraction step is mandatory for discovering the input's most "useful"

TABLE I
The Eight Main Patterns and Their Scansions (Kurta and Kara, 2012)

Pattern	Scansion
Fe'ûlün	■--
Fâ'ilün	-■-
Müstef'ilün	--■-
Mefâ'îlün	■---
Mef'ûlâtü	---■
Fâ'ilâtün	■-■■-
Mütefâ'ilün	■■-■-

TABLE II
The Meters in this Work and their Scansions

Meter	Key pattern	Scansion
Al-Taweel	فعولن مفاعيلن فعولن مفاعلن	o//o// o/o// o/o/o// o/o//
Al-Kamil	متفاعلن متفاعلن متفاعلن	o//o/// o//o/// o//o////
Al-Baseet	مستفعلن فاعلن مستفعلن فعلن	o/// o//o/o/ o//o/ o//o/o/

http://dx.doi.org/10.14500/aro.10631

information, related to the pattern needed to be detected or recognized. For speech inputs, and due to the non-stationary characteristics of the speech, features are normally extracted globally or locally. The local feature is computed for each frame for a length of 30 m, where the signal tends to be more stationary (Rutledge, 1995). The global features are the measurements that represent the whole signal that could be computed by applying some statistics to the local feature values in the frames. In this paper, we extracted 12 LPCCs using LP analysis in addition to 12 MFCCs for each frame of 30 m. A total of 24 features represent each frame, and consequently, the time series version of the data will be represented in terms of a number of *n* samples where each has 24 features for each m_i frame, where i=1, 2,…, n. One of the problems facing time series applications is the unequal length of the input samples that occur where the number of frames per samples is different. In this work, we follow sample frame padding for all of the samples to the length of the longest included sample.

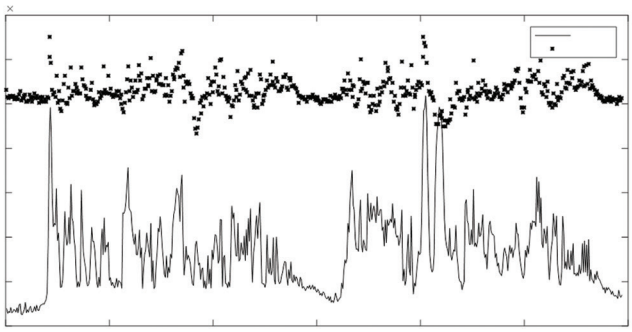

Fig. 1. First and second Mel frequency cepstral coefficient (MFCC) of a sample from the Basset meter. The X-axis represents the number of frames, whereas the Y-axis shows the MFCC value.

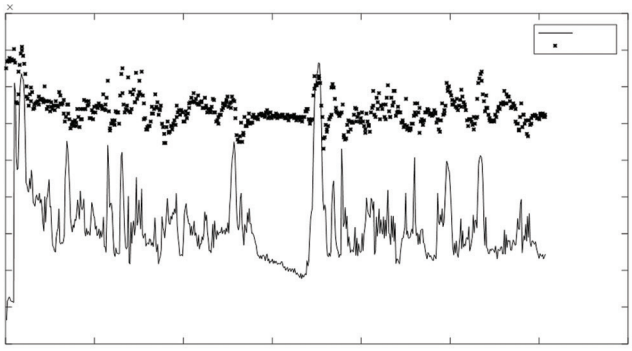

Fig. 2. First and second Mel frequency cepstral coefficient (MFCC) of a sample from the Taweel meter. The X-axis represents the number of frames, whereas the Y-axis shows the MFCC value.

There are other time series classifiers such as the well-known hidden Markov model (HMM). However, the traditional Markov model approaches are limited because their states must be drawn from a modestly sized discrete state space S, and the dynamic programming algorithm is used to perform efficient inference with HMMs scales in time $O(|S|^2)$ (Lipton, et al., 2015). The value of each memory cell in LSTM is controlled with input, modulation, forget, and output gates that allow the LSTM network to store values for many time steps by controlling access to the memory cell (Sønderby, et al., 2015). Consequently, in this work, we propose to use the LSTM classifier for the time series based approach. The LSTM model used in this work includes one LSTM layer with 100 hidden units and Adam optimization, in addition to a fully connected layer followed by a softmax layer and the adopted learning rate is 0.001. The whole recognition process is presented in Fig. 3 and the LSTM architecture is presented in Fig. 4.

For the non-time series version of the data, the mean and the standard deviation of every single feature along the frames have been computed and results in 48 features for every single sample. The features feed a three pairwise SVMs and majority voting is adopted to make the final decision.

The dataset used for this work is prepared and includes 230 Bayt chosen from 10 Arabic poems spoken by two subjects. The data include three meters: Taweel, Kameel, and Baseet with 79, 74, and 77 lines (Bayt).

V. Results and Discussion

The work adopted speaker-dependent (SD) and speaker-independent (SI) approaches to conduct the experiments. In the SD case, 10-fold cross-validation was applied to both time series features, where the LSTM classifier is used, and non-time series features use an SVM classifier. The experiment is applied to each three sets of features, LPCC, MFCC, and in addition to their fusing features. Results show that the MFCC feature achieves better accuracy (0.9957 and 0.9457) than the LPCC feature (0.9939 and 0.9739) using both SVM and LSTM, respectively (Table III). In addition, the fusion features were not able to improve the MFCC accuracy in both models. However, a high diversity

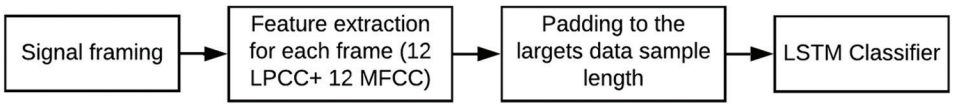

Fig. 3. The adopted algorithm.

Fig. 4. The adopted long short-term memory architecture.

TABLE III
THE AVERAGE AND STANDARD DEVIATION OF SPEAKER DEPENDENT BASED EXPERIMENTS RESULTS OVER 10 FOLDS, "STD" STANDS FOR THE STANDARD DEVIATION

Classifier	Features	Accuracy mean	Accuracy STD	Precision mean	Precision STD	F score mean	F score STD
LSTM	MFCC	0.9457	0.0728	0.9617	0.0482	0.9506	0.0661
	LPCC	0.8834	0.0665	0.9027	0.0631	0.9129	0.0708
	Both	0.9057	0.0838	0.9117	0.0817	0.909	0.0811
SVM	MFCC	**0.9957**	**0.003**	**0.9958**	**0.0031**	**0.9952**	**0.002**
	LPCC	0.9739	0.006	0.9738	0.0052	0.9738	0.0052
	Both	0.9913	0.005	0.9914	0.0041	0.9914	0.0041

MFCC: Mel frequency cepstral coefficient, LSTM: Long short-term memory, SVM: Support vector machine, LPCC: Linear prediction cepstrum coefficient

TABLE IV
THE RESULTS OF THE SPEAKER INDEPENDENT BASED EXPERIMENTS

Classifier	Features	Accuracy	Precision	F score
LSTM	MFCC	**0.8889**	**0.9167**	**0.9026**
	LPCC	0.5111	0.8181	0.6243
	Both	0.6889	0.7520	0.7191
SVM	MFCC	0.6875	0.6875	0.6875
	LPCC	0.6071	0.6071	0.6071
	Both	0.705	0.705	0.705

MFCC: Mel frequency cepstral coefficient, LSTM: Long short-term memory, SVM: Support vector machine, LPCC: Linear prediction cepstrum coefficient

of accuracies among different folds is observed using LSTM in the time series based approach with a standard deviation of 0.0728, 0.0665, and 0.0838 for MFCC, LPCC, and fusion features, respectively. The standard deviation here is a measurement of how diverse the various fold models fit their data. This might reflect the diversity of the phonemes from one Bayt to another in the same meter, where the order of the consonant and vowels is the same. On the other hand, the non-time series model in the SD experiments using SVM records a standard deviation of 0.003, 0.005, and 0.006 for MFCC, LPCC, and fusion features, respectively, as an indication of the stability of different folds, using global feature representation of the data.

It is obvious that in the SD models, the SVM classifier takes a non-time series signal and outperforms the LSTM classifier applied on the time series inputs. This might be an indication that despite the time series nature of the speech input for poetry meter recognition, the non-time series form and signal global features could represent the input data in a significant way. Such an observation of the significance of using the non-time series version of a time series data has been also recorded in other applications like vehicle controlling data (Wells, et al., 2012). However, this performance is limited to SD experiments and not repeated in the SI experiments.

To validate how various speakers influence the meter recognition, we applied SI-based cross-validation. The data in this work include samples recorded by two speakers, one is used to train the model, whereas the other speaker's data are used for model testing. Here, the number of experiments is equal to one because the test and training sets are fixed to the data belonging to each speaker. Experiments with SI show that the same pattern of MFCC outperforms LPCC, and the incapability of the fusion feature to outperform MFCC is repeated also here (Table IV). The most interesting result here is what MFCC+LSTM achieves, where it significantly outperforms other models (0.8889 of accuracy). Time series based MFCC features prove their ability to overcome the SI challenge where the properties of the speaker's voice are not involved in training the model.

A comparison to state-of-the-art studies is inconvenient to some extent due to the nature of the input (audio in the current work and text in all state-of-the-art studies), the number of involved meters (varies from 3 to 16), and the characteristics of the speaker's voice that appears in the SI experiments. However, the proposed SD model using MFCC and SVM achieves an accuracy of 0.9957, which outperforms the best achieved accuracy of 0.9638 in the state-of-the-art studies by Yousef et al. (2019).

VI. CONCLUSION

The Arud meter is a stressed and unstressed pattern in a poem. AAMR is a natural language processing application that aims to determine the type of meter followed in a poem. The Arud meter is traditionally determined from the text of a poem by experts. However, the musical pattern of the poem can also refer to the meter followed in the poem. The main contribution of this paper is to propose a model that automatically recognizes the Arud meter within Arabic poetry using the speech signal of a spoken line (Bayt) of the poem. To the best of our knowledge, no work has used the human voice as an input to AARM. The paper tests both time series and non-time series representations of both MFCC and LPCC features for AARM speech features. The SI results show that MFCCs outperform LPCCs in all of the conducted experiments. The times series based MFCC feature using LSTM detects useful, meter distinguishing information that is shared among the speakers involved. Despite the time series nature of AARM, global non-time series features achieve the best performance; however, this achievement is limited to the SD experiments.

The main deficiencies of this work are the non-adequate variation in the data in terms of number of speakers, number of classes (meters), and in addition to poems in different languages that adopt the same structure of meters. To improve the generalization of the findings in this work, the current exploited data need to be extended to include more samples, speakers, meters, and languages.

REFERENCES

Abuata, B. and Al-Omari, A., 2018. A rule-based algorithm for the detection of arud meter in CLASSICAL Arabic poetry. *International Arab Journal of Information Technology*, 15(4), pp. 1-5.

Al-Falahi, A. Ramdani, M. and Bellafkih, M., 2017, Machine learning for authorship attribution in Arabic poetry. *International Journal of Future Computer and Communication*, 6(2), p. 486.

Almuhareb, A. Alkharashi, I. Al-Saud, L. and Altuwaijri, H., 2013. Recognition of Classical Arabic Poems. In: *2nd Workshop on Computational Linguistics for Literature*, pp. 9-16.

Alnagdawi, M., Rashideh, H. and Aburumman, F., 2013. Finding Arabic poem meter using context free grammar. *Journal of Communication and Computer Engineering*, 3(1), pp. 52-59.

Araújo, L. 2010, Computer-based Assessment (CBA) of Foreign language speaking skills. *JRC Scientific and Technical Reports*, 1, p. 165.

Arberry, J., 1965. *Arabic Poetry. A Primer for Students*. Cambridge University Press, Cambridge.

Hirjee, H. and Brown, D., 2010, Using automated rhyme detection to characterize rhyming style in rap music. *Empirical Musicology Review*, 5(4), pp. 121-145.

Hochreiter, S. and Schmidhuber, J., 1997. Long short-term memory, *Neural computation*, 9(8), pp. 1735-1780.

Ismail, A., Eladawy, M., Keshk, H. and Saleh, S., 2010. Expert system for testing the harmony of Arabic poetry. *Journal of Engineering Sciences*, 1, pp. 401-411.

Kurta, A. and Kara, M., 2012. An algorithm for the detection and analysis of Arud meter in Diwan poetry. *Turk Journal of Electrical Engineering and Computer Science*, 20(6), pp. 948-963.

Lipton, C., Berkowitz, J. and Elkan, C., 2015. *A Critical Review of Recurrent Neural Networks for Sequence Learning, arXiv Preprint arXiv: 1506.00019*. Available from: https://www.arxiv.org/abs/1506.00019.

Morris, H., 1966. On the metrics of pre-islamic Arabic poetry. *Quarterly Progress Report of the Research, Laboratory of Electronics*, 83, pp. 113-116.

Rao, K. and Koolagudi, S., 2013, *Robust Emotion Recognition using Spectral and Prosodic Features*. Springer Science and Business Media. Berlin, Germany, pp. 23-24.

Reynolds, A., 1995. Speaker identification and verification using Gaussian mixture speaker models. *Speech Communication*, 17(1-2), pp. 91-108. Available from: https://www.sciencedirect.com/science/article/abs/pii/016763939500009D.

Rutledge, J.C., 1995. Fundamentals of speech recognition, by lawrence rabiner and bing-hwang juang. *Analysis of Biomedical Engineering*, 23, pp. 526-526.

Sarangi, S.K. and Saha, G., 2020, Improved speech-signal based frequency warping scale for cepstral feature in robust speaker verification system. *Journal of Signal Processing Systems*, 1, 1-14.

Scott, H. 2010. *Pegs, Cords, and Ghuls: Meter of Classical Arabic Poetry*. Swarthmore College Department of Linguistics. Available from: https://www.scholarship.tricolib.brynmawr.edu/handle/10066/6864.

Sønderby, K., Sønderby, K., Nielsen, H. and Winther, O., 2015. Convolutional LSTM Networks for Subcellular Localization of Proteins. *International Conference on Algorithms for Computational Biology, Springer*, pp. 68-80.

Stoetzer, W., 1989. *Theory and Practice in Arabic Metrics.* Leiden, Het Oosters Institute.

Vapnik, V., 1995, *The Nature of Statistical Learning Theory*. Springer-Verlag, New York. Available from: https://www.springer.com/gp/book/9780387987804.

Wells, J.R., Ting, K.M. and Naiwala, C.P., 2012, December. A Non-time Series Approach to Vehicle Related Time Series Problems. Vol. 134. In: *Proceedings of the 10th Australasian Data Mining Conference*, Australian Computer Society, Inc., pp. 61-70.

Yousef, W.A., Ibrahime, O.M., Madbouly, T.M. and Mahmoud, M.A., 2019. Learning meters of Arabic and English poems with Recurrent Neural Networks: a step forward for language understanding and synthesis, *arXiv preprint arXiv:1905.05700*. Available from: https://www.arxiv.org/abs/1905.05700.

Zhang, L. and Gao, J., 2017, A comparative study to understanding about poetics based on natural language processing. *Open Journal of Modern Linguistics*, 7(5), pp. 229-237.