Deep Learning-based Method for Enhancing the Detection of Arabic Authorship Attribution using Acoustic and Textual-based Features

Mohammed Al-Sarem[1], Faisal Saeed[2*], Sultan Noman Qasem[3], Abdullah M Albarrak[4]
College of Computer Science and Engineering, Taibah University, Medina 42353, Saudi Arabia[1]
DAAI Research Group-Department of Computing and Data Science, School of Computing and Digital Technology,
Birmingham City University, Birmingham B4 7XG, UK[2]
Computer Science Department-College of Computer and Information Sciences,
Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh 11432, Saudi Arabia[3, 4]

***Abstract*—Authorship attribution (AA) is defined as the identification of the original author of an unseen text. It is found that the style of the author's writing can change from one topic to another, but the author's habits are still the same in different texts. The authorship attribution has been extensively studied for texts written in different languages such as English. However, few studies investigated the Arabic authorship attribution (AAA) due to the special challenges faced with the Arabic scripts. Additionally, there is a need to identify the authors of texts extracted from livestream broadcasting and the recorded speeches to protect the intellectual property of these authors. This paper aims to enhance the detection of Arabic authorship attribution by extracting different features and fusing the outputs of two deep learning models. The dataset used in this study was collected from the weekly livestream and recorded Arabic sermons that are available publicly on the official website of Al-Haramain in Saudi Arabia. The acoustic, textual and stylometric features were extracted for five authors. Then, the data were pre-processed and fed into the deep learning-based models (CNN architecture and its pre-trained ResNet34). After that the hard and soft voting ensemble methods were applied for combining the outputs of the applied models and improve the overall performance. The experimental results showed that the use of CNN with textual data obtained an acceptable performance using all evaluation metrics. Then, the performance of ResNet34 model with acoustic features outperformed the other models and obtained the accuracy of 90.34%. Finally, the results showed that the soft voting ensemble method enhanced the performance of AAA and outperformed the other method in terms of accuracy and precision, which obtained 93.19% and 0.9311 respectively.**

Keywords*—Authorship attribution; acoustic features; fusion approach; deep learning; CNN; ResNet34

I. INTRODUCTION

As Authorship attribution (AA) refers to the process of identifying the authorship of a document that has not yet been seen, given a list of potential writers and a list of documents that have already been described and tagged by each potential author [1, 2]. AA can be seen as a classification problem, when the training is done on a given set of training texts of known authors (labels). Using a collection of characteristics taken from the text in the training set, we can identify the authors for the testing set. The main characteristics employed during the classifier's training are the style, sentiment, and subject [3]. The authors' individual writing styles are examined in order to extract the stylometry features. Although the writing style of the author might vary depending on the topic, certain persistent and unchecked habits, but the writing styles of the authors cannot be too different over the time [1]. Previously, the authors' attribution was conducted by hand to recognize the authors of unseen texts, but with the huge amount of the texts available online, it is hard to perform this task manually. Several studies addressed this issue in the literature such as [4].

Different statistical and machine learning-based techniques were recently applied on AA [4]. These techniques included Naive Bayes [5, 6], Support Vector Machine (SVM) [7–12], Bayesian classifiers [13], k-nearest neighbor [14, 15], and decision trees [16]. The authorship attribution for texts written in English, Spanish and Chinese has been studied well in the literature; however, less attention was given to the texts written in Arabic because of the complexity of Arabic scripts [17].

To address the Arabic authorship attribution (AAA) issues, several machine and deep learning methods have been applied. For instance, Al-Sarem et al. [17] applied the ensemble machine learning methods with multi-attribute decision making method (TOPSIS) that identifies the base classifier on two Arabic enquires (Fatwa) datasets. The findings showed that AdaBoost and Bagging methods achieved the highest accuracy for the two used datasets. Similarly, Al-Sarem, Alsaeedi, and Saeed [18] applied deep learning-based methods for AAA. The findings showed that the performance of deep learning method outperformed the state-of-art methods.

A recent study by Alqahtani and Dohler [19] reviewed the authorship attribution for Arabic texts. They found that the findings of AAA tasks vary based on the used dataset and features. Also, it was found that few challenges were faced when dealing with pre-processing of Arabic texts because of the scripts' concatenative morphology. For Arabic scripts, all the datasets presented in the previous studies were in texts format and the majority of the currently published studies rely on machine learning techniques. Few studies have previously examined the effectiveness of deep learning for AAA tasks. Therefore, the purpose of this study is to address this gap by

examining how deep learning techniques can be used for enhancing Arabic authorship attribution detection. In addition, to the best of our knowledge, we found that no previous studies investigated the authorship attribution of texts extracted from livestream broadcasting and the recorded Arabic speeches. Therefore, the main contributions of this paper are:

- Scraping data from livestream weekly livestream Arabic sermons as well as the recorded sermons and available publicly on the official website of Al-Haramain in Saudi Arabia (https://www.alharamain.gov.sa/ last access: May 8, 2023).
- Extracting acoustic features and stylometric features directly from the livestream broadcasting and the recorded mp3 files.
- Combining the extracted features using different fusion approaches and Applying different deep learning models for AAA.
- Conducting a rigorous analysis on the applied DL models in terms of different performance evaluation metrics.

This paper is organized as follows: Section II gives an overview of the state-of-the-art methods for detecting authorship attribution. The description of the proposed model's architecture can be found in Section III. Section IV describes the research methodology, including the dataset, preprocessing methods, evaluation metrics, experimental design, and evaluation process. The experimental results were presented in Section V. In Section VI, we summarize the contributions and conclude the paper.

II. Related Studies

Several studies addressed the authorship attribution for Arabic language. For instance, Jambi et al. [20] investigated the feasibility of predicting authorship in Arabic short-microblog content using modern classifiers. To forecast the accuracy of the chosen classifiers, they used three frequently language features—character, lexical, and syntactic—in an incremental approach. Another developing area of machine learning is deep learning, which performs better than classical machine learning in several areas [21] and does not need the use of feature engineering. Although authorship attribution has been a topic of discussion in different languages such as English [22], German [23], Spanish [24], and Chinese [25], only a few studies have specifically addressed the authorship attribution in the context of Arabic [16].

Different deep learning models have been studied in the literature, including the Deep Belief Network (DBN), Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), and Long Short-Term Memory (LSTM). Deep neural networks have recently been used to build authorship attribution systems [26–29]. Usually, input for neural networks takes the form of a list of words or a string of letters. Most techniques focus on lexical aspects, even though lexical-based language models are not scalable when dealing with authorship covering a range of themes [30, 31]. The syntactic features are based on content, more robust, and effective against topic volatility. Sari et al. [26] utilized continuous representation via a neural network together with a classification layer to determine the authorship. In [32], the IMDB62 dataset was used and 94.8% accuracy rate was achieved. According to [30, 33], CNN model was used for authorship identification. They tested their approach using a variety of text data, including emails, reviews, blogs, and tweets, and they were able to get outstanding accuracy scores between 85.0% and 95.0%. The authorship-attribution problem of brief postings was addressed by Shrestha et al. [28] using CNN based on a string of character n-grams. They assessed their methodology using the dataset shown in [34].

To address the issue of authorship attribution, Hitschler et al. [35] used a single author from an anthology reference corpus [36] using POS tags and CNN approach. To achieve proper results from a corpus of scientific papers, they replaced unusual terms with their POS tags in order to create a better generalization. The researchers in [37] used a number of embedding structures based on character, word, n-gram, and POS tags in a CNN model to accurately identify the writing styles used by authors of tweets and posts on Weibo and Twitter. Zhang et al. [29] proposed to encode the syntax parse tree of phrases which contributed towards addressing the authorship attribution problem. By combining lexical characteristics and CNN, they were able to achieve accuracy results of 96.16%, 81.00%, and 56.73% on the IMDB62, CCAT50, and Blogs50 datasets, respectively. Jafariakinabad and Hua [30] used similar datasets and achieved accuracy values of 73.83% and 82.35% on the Blogs50, CCAT50 datasets, respectively, by encoding the syntactic and semantic structures of sentences in texts, and employing a hierarchical neural network based on attention. In [38], a large corpus of blogs with author-provided demographic information was used to study how writing style and content vary by age and gender. It identifies significant differences in vocabulary use and topics among different groups and shows how they can be used for authorship attribution. Moreover, Murauer and Specht [39] applied multiple well-established pre-trained language models to reach better generalization outcomes on the authorship attribution problem. Specifically, they employed variety of language models such as bidirectional encoder representations from Transformers (BERT) [40], a pre-trained general-purpose language representation of DistilBERT [41], and a robustly optimized BERT pre-trained model (RoBERTa) [42] on a large number of extremely diverse authorship-attribution datasets.

Post-authorship attribution is an unresolved research issue because of the numerous inherent uncertainties that text snippets present. On the foundation of a convolutional neural network, a technique for character-level authorship identification was suggested by Modupe et al. [43]. The suggested approach can effectively extract lexical, syntactic, and structural representations from a given post to determine the authorship of dubious materials. Four benchmark experimental datasets were used to compare the performance of this approach to thirteen state-of-the-art approaches.

A character-level RNN was used by Bagnall [44] to model the language of each author in the training set received from the PAN 2015 author verification problem [45]. Additionally, Shrestha et al. [46] used a three-layer CNN model to determine who posted a tweet. The suggested model was used at the

character level since tweets are generally brief in duration. Furthermore, a hierarchical attention-based neural network was also employed by Verma and Srinivasan [47]. The semantic and syntactic structural features are extracted and encoded using the suggested model. Next, document representations were created by combining these features. They used a collection of convolutional layers in their attention-based model to represent a phrase at the word level. While the structural patterns were recorded using an attention based RNN. Moreover, RNN was used by Jafariakinabad et al. [48] to identify a document's syntactic patterns. They used various combinations of DL models to explore both long-term and short-term dependency (POS) tags in phrases. They concluded that the LSTM-based POS are faster than CNN-based POS in capturing the authorial writing style of short texts. Further, Rhodes [49] investigated the effectiveness of the CNN model using a dataset that included 28 English novels written by 14 writers and eight English books written by six authors, both taken from the PAN2012 Authorship Identification competition. The model was used at the sentence level and achieved an accuracy rate of 76% for the books dataset and 20.52% for the PAN2012 English novels.

Ruder et al. [50] examined how different CNN setups affected the impact of attributing various post types. The best performing technique was a character-level CNN, which were compared to other conventional approaches. In addition, a feedforward neural network language model was applied by Ge et al. [51] to train a classifier to attribute a dataset with only a little amount of data. In comparison to n-gram baselines, the model trained a representation for each word using a window of four grams and achieved an accuracy of 95%.

According to the conducted review of the literature, there are few state-of-the-art techniques for identifying Arabic language writers, especially for the text extracted from livestream broadcasting and the recorded speeches. Most of the existing studies investigated the machine learning methods for AAA. Therefore, this study aims to investigate how to apply deep learning for solving Arabic authorship attribution using religious texts extracted from livestream broadcasting and recorded speeches. The dataset was collected from the official website of Al-Haramain for several authors (scholars) in Saudi Arabia.

III. PROPOSED APPROACH

As mentioned earlier, this paper aims to enhance the performance of the authorship attribution detection. For this purpose, a combination of acoustics and stylometric features were extracted and combined, and deep learning methods were applied. Generally, the proposed approach consists of the following stages: feature extraction, data preprocessing, detection and classification, and ensemble learning-based techniques.

Fig. 1 shows the overview of the proposed approach used for extracting the required features and applying the classification. The full description of the proposed approach stages are given in the subsections.

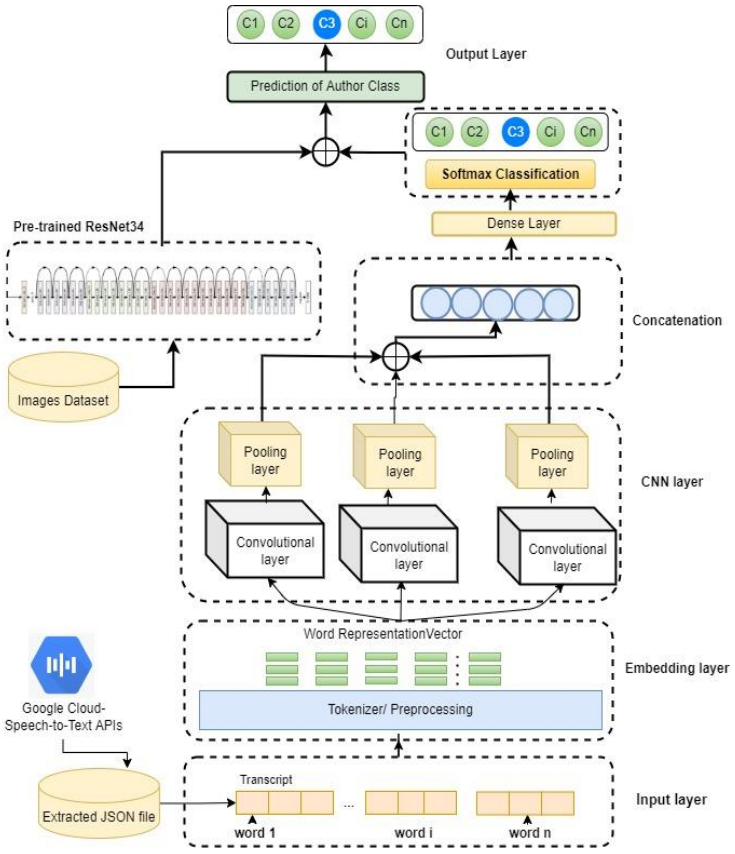

Fig. 1. The proposed approach for detecting the class of authors.

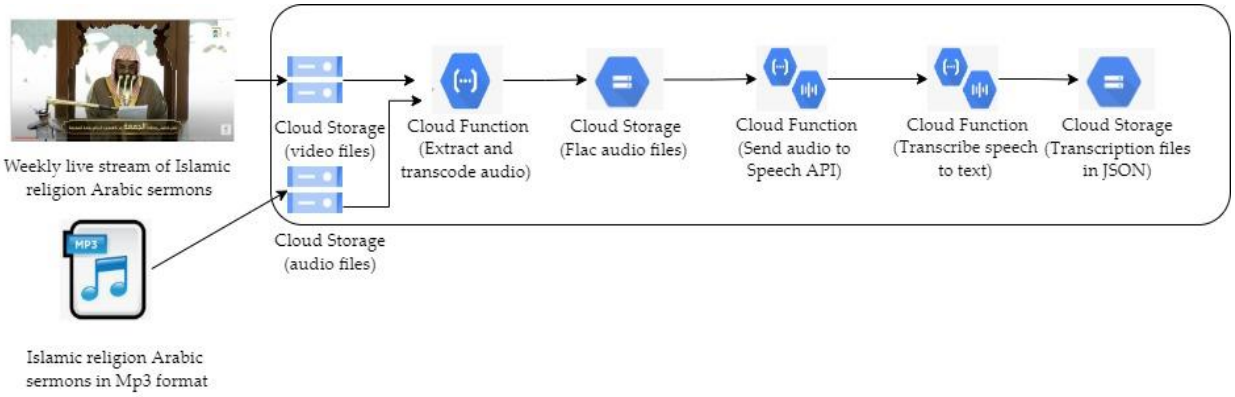

Fig. 2. Transcription speech to text within Google Cloud storage. The process shows two scenarios: (i) the video files and (ii) audio files.

A. Extraction Stage

In this stage, a set of acoustic features, textual and stylometric features are extracted from the weekly live stream of Arabic sermons and the recorded speeches in mp3 files that are available publicly on the official website of Al-Haramain. We collected the dataset from this source because the Modern Standard Arabic (Fusha) is used in all sermons' texts and recorded speeches. The google cloud APIs were used to extract audio files from the livestream videos and to transcribe speech to text.

The final transcription is stored in JSON format and then passed to the next stages. Fig. 2 shows the process of extracting the acoustics and stylometric features using google cloud APIs. The process of extracting the raw textual data from the audio file after converting it to FLAC format is summarized in Algorithm 1. Also, Table I shows a fragment of JSON file obtained as the output of the Google cloud API and the JSON request file fired in Google cloud. For determining the SampleRateHertz of an audio file, we used Apache Tika tool[1].

Algorithm 1: Extraction textual data from audio files

Input: livestream Arabic sermons ArS ; recorded mp3 files $AudioF$

Output: JSON file

```
Initialize: FileMetadata= [] // empty list
For (every ArS and AudioF) do
    ConvertToFlac( )  // convert file to FLAC format
    FileMetadata ← GetMetadata() // get file metadata including
                        // SampleRateHertz by
                        // Apache Tike tool
    JSON_file = ConstructJSON (FileMetadata)
    End
Return JSON_file
```

B. Data Preprocessing

1) Preprocessing the textual data: Before feeding the raw textual data into the proposed DL model, a set of preprocessing techniques were applied. This is a necessary step to enhance the quality of the DL model. AL-Sarem et al. [52] investigated the importance of the preprocessing techniques in increasing the performance of ML models. The preprocessing phase includes data cleansing techniques, stemming and tokenization techniques. In terms of the used cleansing techniques, since the textual data is obtained directly from the JSON files that were generated by the Google speech-to-text APIs, the only necessary steps are fixing the spelling errors, stop word removal, and completing the words that might truncated due to the non-complete video frame splitting (this is because the video frames had a fixed length and we set them to 30 second per a frame). In addition, we performed the stemming process using the porter stemmer as recommended in [53].

2) Preprocessing the audio files: We treated the audio files as the source for obtaining the acoustic features such as waveforms, spectrogram, spectral roll-off and chroma features. The audio waveforms are represented as spectrograms, which depict the intensity of a signal over time at various frequencies. Fig. 3 to 6 show a sample of waveform, spectrogram, spectral roll-off and chroma images that were obtained when we passed the same sliced audio signal to the acoustics feature extractor.

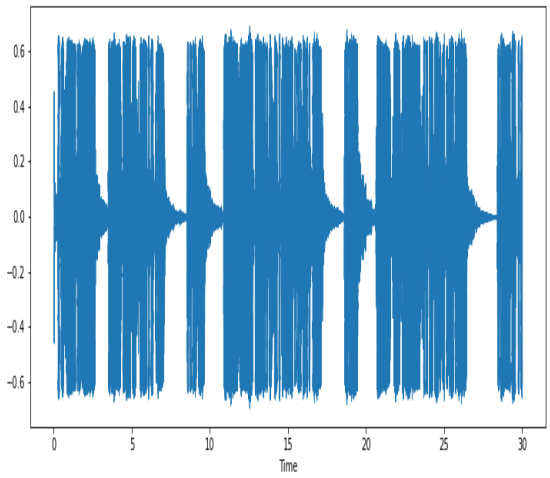

Fig. 3. Wave form that was obtained from the sliced audio signal.

[1] https://tika.apache.org/

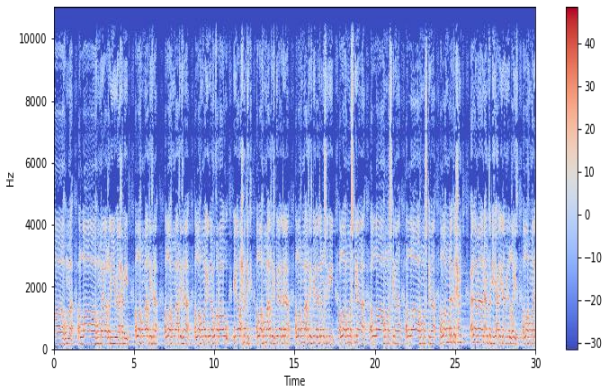

Fig. 4. Spectrogram that was obtained from the sliced audio signal.

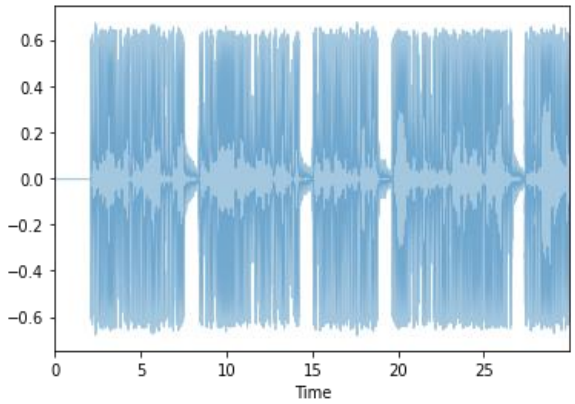

Fig. 5. Spectral roll-off that was obtained from the sliced audio signal.

TABLE I. JSON FILES: THE LEFT COLUMN PRESENTS A FRAGMENT OF JSON REQUEST FILE, MIDDLE COLUMN PRESENTS THE JSON OBTAINED FILE, AND THE LAST COLUMN A TRANSLATION OF THE OBTAINED TRANSCRIPT TO ENGLISH

JSON File Request	JSON file Obtained	Translation to English
{ "config": { "encoding":"FLAC", "sampleRateHertz": 44100, "Language Code": "ar-SA", "enableWordTimeOffsets": false }, "audio": { "uri":"~/audio.flac" }}	{"results":[{ "alternatives":[{ "transcript": " و الأخرة خير لمن اتقى ولا تظلمون فتيلا وان كل ذلك لما متاع الحياه الدنيا والأخرة عند ربك للمتقين فاحذر ايها المسلم من الإغترار بهذه الدنيا وكذا الوقاية من الغفلة والشهوة والهوى قال جل وعلى يا ايها الناس ان وعد الله حق ", "confidence": 0.98267895 }] }]}	"Say, ˊO Prophet, "The enjoyment of this world is so little, whereas the Hereafter is far better for those mindful ˊof Allahˋ. And none of you will be wronged ˊeven by the width of" the thread of a date stone. Beware, O Muslim, from being deceived by this world, as well as avoiding negligence, lust, and whims. God Almighty said, O people, God's promise is true."

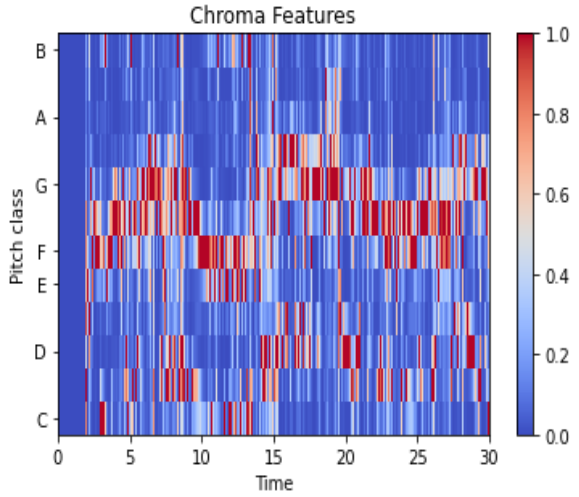

Fig. 6. Chroma that was obtained from the sliced audio signal.

C. Detection and Classification Stage

For the detection and classification stage, the textual data that were extracted as JSON file by the google speech-to-text APIs and prepared at the second stage were fed into the deep learning-based models. DL-based techniques possess the main advantage that no feature engineering is needed [54, 55]. In the training phase, DL classifiers extract and obtain useful features from the input data directly. In this section, we described the models that were applied in this study such as CNN architecture and its pre-trained ResNet34. In addition, this section explains how we used multi-input word embedding as a text representation. For the acoustic features, the pretrained ResNet34 model was fine-tuned and trained using the spectrogram images dataset obtained from the audio files. Then, the outputs of both models (multi-input CNN-based models and the pre-trained ResNet34 model) were combined. Finally, different ensemble methods were used to obtain more accurate results.

1) Multi-input word embedding model: The concept of word embedding (WE) refers to a technique of a text representation in which words having the same meaning are represented similarly. *Word2vec*, *GloVe*, and *FastText* are some recent word-embeddings that are commonly used with ML and DL models. In the proposed multi-input channel CNN model, we used the *Keras* embedding layer. The proposed model processes textual data in three channels with four, six, and eight grams of input. The word-embedding vector has an output dimensions size of 100 with a maximum length of sequences (input length) computed directly from the input textual data.

2) Architecture of the proposed CNN model: As shown in Fig. 1, the word embedding layer was connected to three parallel CNN blocks. Each block (channel) composes of the following:

- The length of input sequences.
- A layer of embedding set with 100-dimensional real-valued representations.
- A convolution layer with 32 filters and a kernel size set to the number of words to be read simultaneously.
- Max Pooling layer to consolidate the output from the convolutional layer.
- Flatten layer to reduce the dimensional output before concatenation phase.

Finally, we concatenated the output from the three channels into a single vector and processed it through a dense layer and an output layer. Table II summarizes the configuration of each CNN block.

TABLE II. CNN LAYERED ARCHITECTURE

CNN Block	Output Dimension	Kernal Size	Filter	Dropout Rate	Activation Function
Block No. 1	100	4	32	0.5	relu
Block No. 2	100	6	32	0.5	relu
Block No. 3	100	8	32	0.5	relu

3) ResNet34 model: ResNet34 model is an image-based pre-trained CNN model with 34 convolution layers, one *MaxPool*, and one average pool layer. Unlike the CNN-based models which commonly experience vanishing gradients during backpropagation, the skipping connections in the ResNet34 [56] are used to solves this issue.

As shown in Fig. 7, the first convolutional layer in ResNet34 model has filter with kernel size of 7x7 followed by a *MaxPool* layer (the stride is set to 2, which is indicated as "/2" in the Figure). Next, a group of convolution layers were connected using skip connection which is jumping every two layers. The layer of the first group (colored in grey) has kernel size of 3×3 and filters of 64. This layer is repeated three times and layered between the *MaxPool* layer and the layers of the second group. The layers of the second group have filter of 128 and *Kernal* size of 3x3 and these are get repeated on this time four times keeping the same skip connection length. In this manner, we continue until we reach the *avg_pooling* and *softmax* functions. Each time, the number of filters gets doubled.

4) Ensemble learning-base fusion strategies: Ensemble learning is a technique for merging the outputs of different ML models to improve the overall performance [57-59]. The empirical results show that when different ML/DL models work together, the performance is improved compared to the performance of standalone single model. Armed with this concept, two ensemble learning strategies: the hard voting and soft voting were implemented to combine the output of the multi-input channel CNNs models and the ResNet34 model.

a) Hard voting approach: The hard voting approach follows the majority voting concept in which a class with the most n votes is considered as the final output class. In general, each classifier $c \in \mathbb{C}$, makes its own prediction, and a vector size of n store the results, where n is the number of classifiers that participate in voting pool: $[C_1(x), C_2(x), \ldots, C_n(x)]$. The output class y, is then, predicted by applying the formula presented in Equation (1).

$$y = mode[C_1(x), C_2(x), \ldots, C_n(x)] \quad (1)$$

b) Soft voting approach: Unlike the hard voting, soft voting determines the output class by projecting probability p of all classifiers [57]. Then, the average probability is computed for each class as follows:

$$P_{max}(i_n|x) = \frac{1}{n}\sum_{k=1}^{n} Pm_k(i_j|x)$$

$$Y = argmax[Pmean(i0|x), \ldots, Pmean(ij|x)]$$

Then, taking into consideration the greatest probability, the output class y is determined according to the formula in Equation (2) as follows:

$$y = argmax[P_{max}(i_0|x), P_{max}(i_1|x), \ldots, P_{max}(i_n|x)] \quad (2)$$

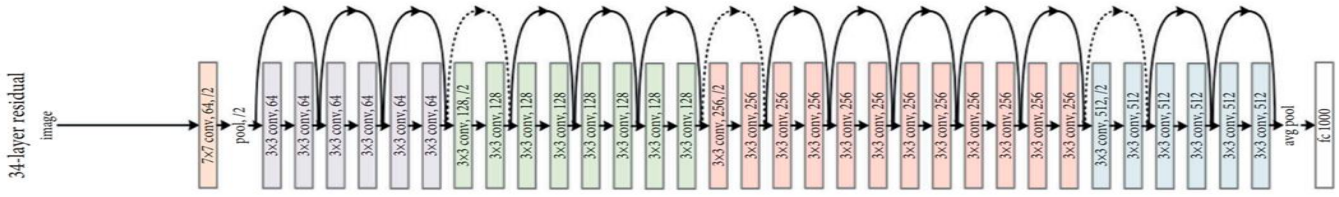

Fig. 7. ResNet34 model architecture, "skip connection" among the layers was depicted as a black curved arrow.

IV. EXPERIMENTS

This section presents and analyses the finding results of the proposed model and the ensemble learning-based fusion techniques that applied to dataset of textual and acoustic-based features.

A. Experimental Setup

In this study, an Intel(R) Core (TM) i9-10980HK CPU @ 2.40 GHz and an NVIDIA GeForce MX graphics card run on Windows 11 with 32 GB RAM to implement the proposed methods. All the tested DL models were encoded using Python 3.9 using the *Jupyter* Notebook (available online: https://jupyter.org). It is also provided by Anaconda distribution (available online: https://www.anaconda.com). In

addition, we employed the open-source *PyTorch* library (available online: https://pytorch.org) to implement the ResNet34 and tune the multi-input channel CNN models. Since the main aim of the proposed model is to enhance the performance of classifying the authors of Arabic scripts and live stream or recorded Arabic sermons, a set of different DL architectures over 35 epochs was trained. Later, we compared the performance of these models to the proposed model. In addition, the models applied the Adam optimizer and the cross-entropy loss function at 1e-3 learning rate. The Spectrogram images of the audio signal with size of (224 × 224) pixels and batch size of 64 were used for training the ResNet34. Table III summarizes the hyperparameters used for models training.

TABLE III. CONFIGURATION PARAMETERS OF THE RESNET34

Parameters	Values
Epochs	35
Batch Size	64
Learning rate	1e-3
Optimizer	Adam Optimizer
Spectrogram image size	(224x224)
Loss function	Cross-entropy

B. Dataset

As mentioned earlier, the dataset was scrapped directly from the official website of Al-Haramain in Saudi Araiba (https://www.alharamain.gov.sa/) and its official YouTube channel (https://www.youtube.com/watch?v=o5tC9aWaQ80/).

The size of scrapped data was about 2 GB. Table IV shows some statistical characteristics of the dataset. Since the dataset is imbalanced in terms of number of videos (see Fig. 8), we assumed a Gaussian distribution over the candidate authors.

In addition, the dataset contains 14,912 spectrogram images size of (224x224) representing five distinct classes. In the experimental part, we split the dataset into 70% for training, 20% for validation, and 10% for testing.

TABLE IV. SOME STATISTICAL CHARACTERISTICS OF DATASET

Author	No. of videos	Average video length in (Min)	Total Size in (MB)
AAl AlShaikh	42	18.24	344
Ahmed Hameed	20	15.38	153
Albaejan	43	17.32	308
Alhudaify	68	24.55	775
AlQaseem	49	20.46	482

C. Performance Metrics

For any classification problem, the performance of ML and DL models can be evaluated by computing the classification accuracy, precision, recall, and F1 score. In this paper, to precisely assess the proposed method, the experiments were conducted and validated using 5-fold cross-validation method. The confusion matrix (refer to Table V for more details) is presented to demonstrate how the statistical metrics (accuracy, precision, recall, and f1-score) can be computed.

$$Accuracy = \frac{TP+TN}{TN+FP+FN+TP} \quad (3)$$

$$Precision = \frac{TP}{TP+FP} \quad (4)$$

$$Recall = \frac{TP}{TP+FN} \quad (5)$$

$$F1-score = \frac{2*Precision*Recall}{(Precision+Recall)} \quad (6)$$

TABLE V. CONFUSION MATRIX

	Predicted Negative	Predicted Positive
Actual Negative	True negative (TN)	False positive (FP)
Actual Positive	False negative (FN)	True positive (TP)

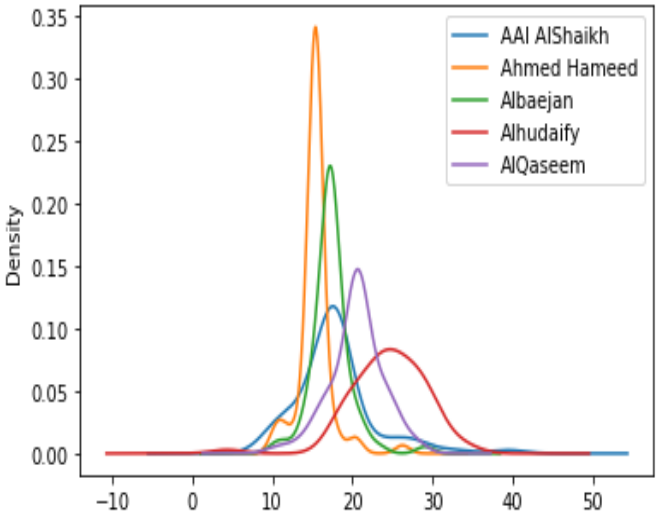

Fig. 8. Distribution of video length as per an author class.

V. RESULTS

The subsections below summarize the findings of the experiments performed on the used dataset. We first presented the results obtained by employing only the extracted textual data. Then, the performance of ResNet34 model was highlighted. Later, we showed the results of fusing both features using the two fusion strategies. It is important here to mention that the results presented in the following subsections are the average value of each experiment that was repeated five times independently.

A. Evaluation with Textual Data

To show the impact of the proposed CNN model with multi-input channels, different kernel size input layer was implemented. Table VI shows the performance of the CNN model with different input channels. The results of CNN model with the multi-input channel are highlighted at the bottom of the table. In addition, the structure of the proposed CNN model with multi-input channel is depicted on Fig. 9.

The results show that the use of textual data yields an acceptable performance in terms of all measuring metrics. However, when the CNN model was restructured and the multi-input blocks were added to the original CNN model, a notable improvement was observed in terms of all metrics which encourage us to keep this structure and test it after the fusion with ResNet34 model.

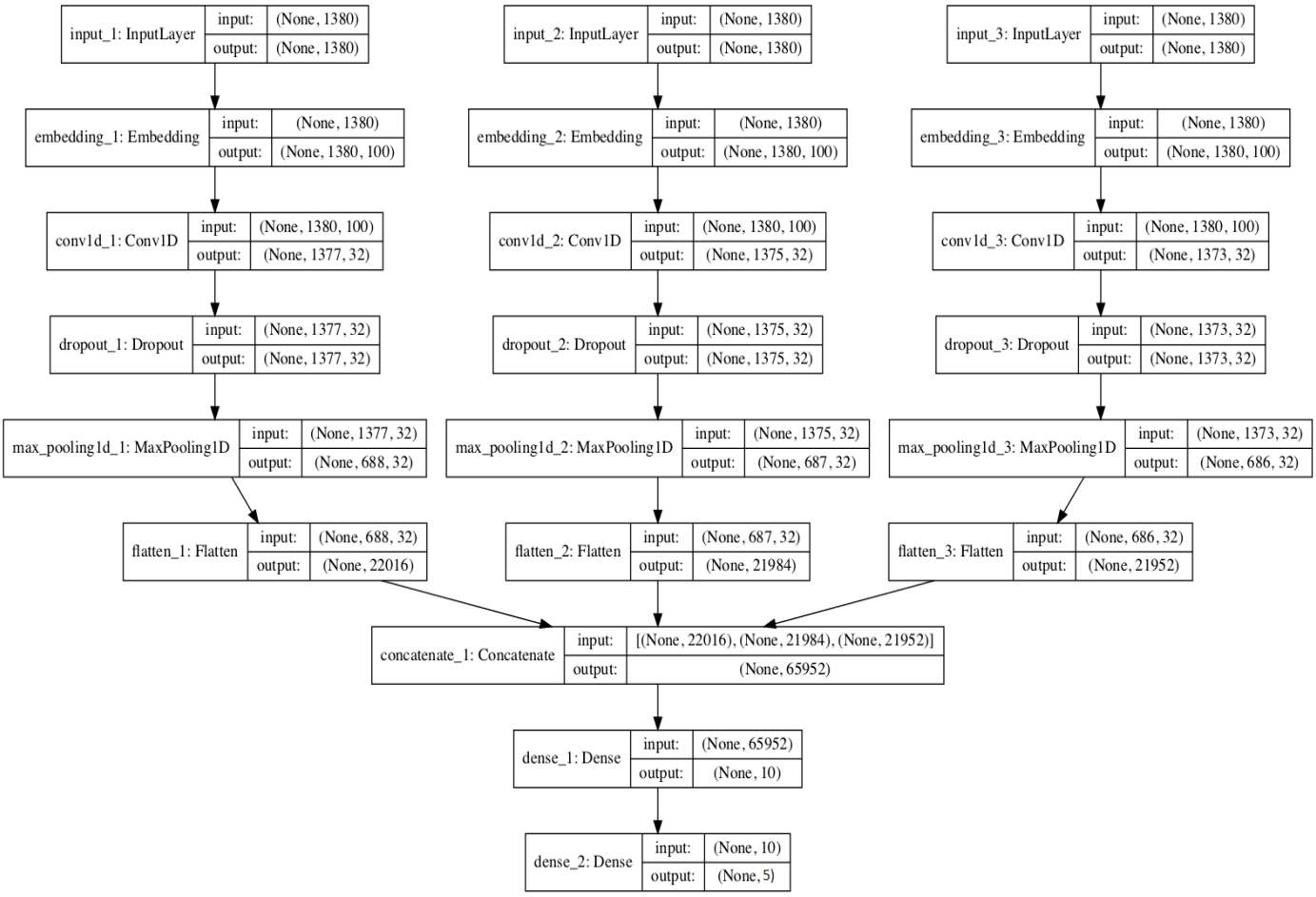

Fig. 9. Architecture of the proposed CNN model with the multi-input channel.

TABLE VI. THE PERFORMANCE OF CNN MODELS WITH DIFFERENT INPUT CHANNELS

CNN model	Accuracy	Precision	Recall	F1-Score
Block No. 1	85.26%	0.856	0.853	0.853
Block No. 2	85.10%	0.851	0.851	0.851
Block No. 3	85.42%	0.856	0.854	0.854
The proposed Model	**86.57%**	**0.864**	**0.861**	**0.861**

B. Evaluation with Spectogram Images

The acoustic features that were extracted from the audio files are fed into the ResNet34 model. As we mentioned earlier, several acoustic features can be extracted as images from each audio file such as waveforms, spectrogram, spectral roll-off and chroma features. The spectrogram-based images were used. To illustrate the performance of *ResNet34*, two other pretrained DL models were also implemented namely *Xception* Model [60] and VGG-19 model [61]. Table VII shows the performance of the deployed DL models achieved when the spectrogram images were used as a training set. The results show that the *ResNet34* model overcomes other models and yields accuracy of 90.34%, and 0.903, 0.899, and 0.901 in terms of precision, recall, and F1-score respectively.

TABLE VII. PERFORMANCE OF CNN-BASED MODELS USED IN THIS PAPER

CNN model	Accuracy	Precision	Recall	F1-Score
VGG19	79.12%	0.789	0.801	0.795
Xception	85.31%	0.850	0.860	0.850
ResNet34	**90.34%**	**0.903**	**0.899**	**0.901**

C. Fusion Model

In the previous subsections, the experimental results proved superiority of multi-input channel CNN model and ResNet34 model in detection and classification the authors' class using the textual data and the acoustic-based features. The next step is to examine the impact of fusing both models using various learning strategies. Both the hard and soft voting approaches have been investigated. Table VIII presents the performance results of the proposed model with respect to the used ensemble learning strategies.

TABLE VIII. PERFORMANCE RESULTS OF THE PROPOSED MODEL WITH RESPECT TO THE ENSEMBLE LEARNING STRATEGIES

Ensemble Learning	Accuracy	Precision	Recall	F1-score
Hard voting	92.75%	0.9258	**0.9287**	0.9272
Soft voting	**93.19%**	**0.9311**	0.9271	**0.9291**

In addition, the results show the superiority of the combined models over the individual models. The results show that soft voting approach clearly outperforms the hard voting approach in terms of accuracy and precision and both approaches obtained similar results in terms of recall and F1-score. As a result, the soft voting approach is the recommended assembling strategy.

VI. CONCLUSION

This study proposed an Arabic authorship attribution approach that includes five main stages: feature extraction, data preprocessing, detection and classification, and ensemble learning-based fusion strategies. In the first stage, a set of acoustic, textual and stylometric features were extracted from different Arabic live stream sermons and recorded Arabic speeches files for five authors. In the data preprocessing, several techniques were applied for data cleansing, stemming and tokenization. The audio waveforms were represented as spectrograms, which depict the intensity of a signal over time at various frequencies. Next, in the detection and classification stage, the extracted data were fed into the deep learning-based models (CNN architecture and its pre-trained ResNet34). Then, hard and soft voting ensemble methods were applied for combining the outputs of the applied models to improve the overall performance. The experimental results showed that the use of textual data with CNN yields an acceptable performance in terms of all evaluation metrics. Then, when the acoustic features were extracted from the audio files and fed into the ResNet34 model, the results show that the ResNet34 model overcomes other models and yields accuracy of 90.34%, and 0.903, 0.899, and 0.901 in terms of precision, recall, and F1-score respectively. Finally, when the outputs of the multi-input channel CNNs models and the ResNet34 model were combined using hard and soft voting ensemble methods, the results show the superiority of the combined models over the individual ones. The results show that soft voting approach clearly outperforms the hard voting approach in terms of accuracy and precision (93.19% and 0.9311 were obtained respectively). Future works can investigate the application of the proposed model on different datasets and apply different CNN architectures and pre-train models. Different fusion methods can be applied and lead to obtain more enhancements for the detection of AAA.

ACKNOWLEDGMENT

The authors extend their appreciation to the Deputyship for Research & Innovation, Ministry of Education in Saudi Arabia for funding this research work through project number IFP-IMSIU202106.

FUNDING

This research is funded by the Deputyship for Research & Innovation, Ministry of Education in Saudi Arabia through project number IFP-IMSIU202106.

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

REFERENCES

[1] M. Al-Sarem and A.H. Emara, “The effect of training set size in authorship attribution: application on short arabic texts,” International Journal of Electrical and Computer Engineering ;9(1), 652, 2019.

[2] L. Srinivasan and C. Nalini, “An improved framework for authorship identification in online messages,” Cluster Computing, 1-10, 2017.

[3] N. Potha and E. Stamatatos, “Improved algorithms for extrinsic author verification, Knowledge and Information Systems,” 1-19, 2019.

[4] E. Stamatatos, “A survey of modern authorship attribution methods,” Journal of the American Society for information Science and Technology, 60(3), 538-556, 2009.

[5] A. S. Altheneyan and M. E. B. Menai, “Naïve Bayes classifiers for authorship attribution of Arabic texts,” Journal of King Saud University-Computer and Information Sciences, 26(4), 473-484, 2014.

[6] G. Baron, “Influence of data discretization on efficiency of Bayesian classifier for authorship attribution,” Procedia Computer Science, 35, 1112-1121, 2014.

[7] J. P. Posadas-Durán, H. Gómez-Adorno, G. Sidorov, I. Batyrshin, D. Pinto, and L. Chanona-Hernández, “Application of the distributed document representation in the authorship attribution task for small corpora,” Soft Computing, 21(3), 627-639, 2017.

[8] L. Pan, I. Gondal, and R. Layton, “Improving Authorship Attribution in Twitter Through Topic-Based Sampling, Lecture Notes in Computer Science, 10400. Springer, Cham, 2017.

[9] E. Dauber, R. Overdorf, and R. Greenstadt, “Stylometric Authorship Attribution of Collaborative Documents,” In International Conference on Cyber Security Cryptography and Machine Learning, 115-135. Springer, Cham, 2017.

[10] O. Marchenko, A. Anisimov, A. Nykonenko, T. Rossada, and E. Melnikov, “Authorship Attribution System.” Lecture Notes in Computer Science, vol 10260. Springer, Cham, 2017.

[11] F. Claude, D. Galaktionov, R. Konow, S. Ladra, and Ó. Pedreira, “Competitive Author Profiling Using Compression-Based Strategies,” International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 25(Suppl. 2), 5-20, 2017.

[12] A. Al-Falahi, M. Ramdani, and B. Mostafa, “Machine Learning for Authorship Attribution in Arabic Poetry,” International Journal of Future Computer and Communication, 6(2), 42, 2017.

[13] G. Baron, “Influence of data discretization on efficiency of Bayesian classifier for authorship attribution,” Procedia Computer Science, 35, 1112-1121, 2014.

[14] P. P Paul, M. Sultana, S. A. Matei, and M. Gavrilova, “Authorship disambiguation in a collaborative editing environment,” Computers & Security, 2018.

[15] C. Akimushkin, D. R. Amancio, and O. N. Oliveira, ”On the role of words in the network structure of texts: application to authorship attribution,” Physica A: Statistical Mechanics and its Applications, 495, 49-58, 2018.

[16] M. Al-Sarem and A. H. Emara, “Analysis the Arabic Authorship Attribution Using Machine Learning Methods: Application on Islamic Fatwā,” In International Conference of Reliable Information and Communication Technology, (pp. 221-229). Springer, Cham, 2018.

[17] M. Al-Sarem, F. Saeed, A. Alsaeedi, W. Boulila, and T. Al-Hadhrami, “Ensemble methods for instance-based arabic language authorship attribution,” IEEE Access, 8, pp.17331-17345, 2020.

[18] M. Al-Sarem, A. Alsaeedi, and F. Saeed, “A deep learning-based artificial neural network method for instance-based arabic language authorship attribution,” Int J Adv Soft Comput Appl, 12(2), 1-15, 2020.

[19] F. Alqahtani and M. Dohler, “Survey of Authorship Identification Tasks on Arabic Texts,” Transactions on Asian and Low-Resource Language Information Processing, 2022.

[20] K. M. Jambi, I. H. Khan, M.A. Siddiqui, and S.O. Alhaj, “Towards Authorship Attribution in Arabic Short-Microblog Text,” IEEE Access, 9, 128506-128520, 2021.

[21] M. Al-Sarem, W. Boulila, M. Al-Harby, J. Qadir, and A. Alsaeedi, "Deep Learning-Based Rumor Detection on Microblogging Platforms: A Systematic Review," IEEE Access, 7, 152788-152812, 2019.

[22] D. Labbé, "Experiments on authorship attribution by intertextual distance in English," Journal of Quantitative Linguistics, 14(1), 33–80, 2017.

[23] J. Savoy, "A comparative study of three text corpora and three languages," J. Quant. Linguist, 19(2), 132–161, 2012.

[24] R. Oppliger, "Automatic authorship attribution based on character n-grams in Swiss German," In: Proceedings of the 13th Conference on Natural Language Processing (KONVENS 2016), 2016.

[25] M. Crespo and A. Frías, "Stylistic authorship comparison and attribution of Spanish news forum messages based on the Tree Tagger POS Tagger," In: 33rd Conference of the Spanish Association of Applied Linguistics (AESLA), XXXIII AESLA Conference, Madrid, Spain, 16–18 April 2015

[26] Y. Sari, A. Vlachos, and M. Stevenson, "Continuous n-gram representations for authorship attribution," In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, Valencia, Volume 2, pp. 267–273. Spain, 3–7, April 2017.

[27] S. Ruder, P. Ghaffari, and J. G. Breslin, "Character-level and multi-channel convolutional neural networks for large-scale authorship attribution," arXiv, arXiv:1609.06686, 2016.

[28] P. Shrestha, S. Sierra, F.A. González, M. Montes-y-Gómez, P. Rosso, and T. Solorio, "Convolutional neural networks for authorship attribution of short texts," In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics EACL, pp. 669–674, Valencia, Spain, 3–7 April 2017.

[29] R. Zhang, Z. Hu, H. Guo, and Y. Mao, "Syntax encoding with application in authorship attribution," In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pp. 2742–2753, Brussels, Belgium, 31 October–4 November 2018.

[30] F. Jafariakinabad and K. A Hua, "Style-Aware Neural Model with Application in Authorship Attribution," In Proceedings of the 2019 18th IEEE International Conference on Machine Learning and Applications (ICMLA), pp. 325–328, Boca Raton, FL, USA, 16–19 December 2019.

[31] F. Jafariakinabad, S. Tarnpradab, and K. A. Hua, "Syntactic neural model for authorship attribution," In Proceedings of the Thirty-Third International Flairs Conference, pp. 234–239, Miami, FL, USA, 17–18 May 2020.

[32] Y. Seroussi, I. Zukerman, and F. Bohnert, "Authorship attribution with latent Dirichlet allocation," In Proceedings of the Fifteenth Conference on Computational Natural Language Learning, pp. 181–189, Portland, OR, USA, 23–24 June 2011.

[33] Y. Kim, "Convolutional neural networks for sentence classification," In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP) pp. 1746–1751, Doha, Qatar, 25–29 October 2014.

[34] R. Schwartz, O. Tsur, A. Rappoport, and M. Koppel, "Authorship attribution of micro-messages," In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pp. 1880–1891, Seattle, WA, USA, 18–21 October 2013.

[35] J. Hitschler, E. van den Berg, I. Rehbein, Authorship attribution with convolutional neural networks and POS-Eliding, In Proceedings of *the Workshop on Stylistic Variation*, pp. 53–58. Copenhagen, Denmark, 8 September 2017.

[36] S. Bird, R. Dale, B. J. Dorr, B. Gibson, M. T. Joseph, M. Kan, D. Lee, B. Powley, D. R. Radev, and Y. F. Tan, "The ACL anthology reference corpus: A reference dataset for bibliographic research in computational linguistics," In Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08), Marrakech, Morocco, 28–30 May 2008.

[37] Z. Hu, R. K.-W. Lee, L. Wang, E. Lim, B. Dai, "Deepstyle: User style embedding for authorship attribution of short texts," In Proceedings of the Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint International Conference on Web and Big Data, pp. 221–229, Tianjin, China, 12–14 August 2020.

[38] J. Schler, M. Koppel, S. Argamon, and J. W. Pennebaker, "Effects of age and gender on blogging," In Proceedings of the AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs, Volume 6, pp. 199–205, Stanford, CA, USA, 27–29 March 2006.

[39] B. Murauer and G. Specht, "Developing a benchmark for reducing data bias in authorship attribution," In Proceedings of the 2nd Workshop on Evaluation and Comparison of NLP Systems, pp. 179–188, Punta Cana, Dominican Republic, 10–11 November 2021.

[40] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language under-standing," arXiv, arXiv:1810.04805, 2018.

[41] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter," arXiv, arXiv:1910.01108, 2019.

[42] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized Bert pretraining approach," arXiv, arXiv:1907.11692, 2019.

[43] A. Modupe, T. Celik, V. Marivate, and O. O. Olugbara, "Post-Authorship Attribution Using Regularized Deep Neural Network," Applied Sciences. 12, 7518. 2022.

[44] D. Bagnall, "Author Identification using multi-headed Recurrent Neural Networks," Notebook for PAN at CLEF 2015 Evaluation Labs and Workshop, Toulouse, France, 8-11 September, 2015.

[45] E. Stamatatos, W. Daelemans, B. Verhoeven, M. Potthast, B. Stein, P. Juola, M. Sanchez-Perez, and A. Barrón-Cedeño, "Overview of the author identification task at PAN 2014," In CLEF 2014 Evaluation Labs and Workshop Working Notes Papers, Sheffield, UK,; pp. 1-21, 2014.

[46] P. Shrestha, S. Sierra, F. Gonzalez, M. Montes, P. Rosso, and T. Solorio, "Convolutional neural networks for authorship attribution of short texts," In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers, pp. 669-674, 2017.

[47] G. Verma and B. V. Srinivasan, "A Lexical, Syntactic, and Semantic Perspective for Understanding Style in Text," arXiv, 2019.

[48] F. Jafariakinabad, S. Tarnpradab, and K. A. Hua, "Syntactic Recurrent Neural Network for Authorship Attribution," arXiv, arXiv:1902.09723, 2019.

[49] D. Rhodes, "Author Attribution With CNNs," Avaiable online: https://www. semanticscholar. org/paper/AuthorAttribution-with-Cnn-s-Rhodes/0a904f9d6b47dfc574f681f4d3b41bd840871b6f/pdf (accessed on 22 August 2016) (2015).

[50] S. Ruder, P. Ghaffari, and J. G. Breslin, "Character-Level and Multi-Channel Convolutional Neural Networks for Large-Scale Authorship Attribution," arXiv, arXiv:1609.06686, 2016.

[51] Z. Ge, Y. Sun and M. J. T. Smith, "Authorship Attribution Using a Neural Network Language Model," In Proceedings of the AAAI Conference on Artificial Intelligence, 4212–4213, 2016.

[52] M. Al-Sarem, M. Al-Harby, F. Saeed, and E. A. Hezzam, "Machine learning classifiers with pre-processing techniques for rumour detection on social media: an empirical study," International Journal of Cloud Computing, 11(4), 330-344, 2022.

[53] A. Alsaeedi and M. Al-Sarem, "Detecting Rumors on Social Media Based on a CNN Deep Learning Technique," Arabian Journal for Science and Engineering, 45, 10813–10844, 202

[54] M. Al-Sarem, A. Alsaeedi, F. Saeed, W. Boulila, and O. A. AmeerBakhsh, "Novel Hybrid Deep Learning Model for Detecting COVID-19-Related Rumors on Social Media Based on LSTM and Concatenated Parallel CNNs," Applied Sciences, 11, 7940, 2021.

[55] O. El Gannour,S. Hamida, B. Cherradi, M. Al-Sarem, A. Raihani, F. Saeed, and M. Hadwan, "Concatenation of Pre-Trained Convolutional Neural Networks for Enhanced COVID-19 Screening Using Transfer Learning Technique," Electronics, 11, 103, 2022.

[56] M. Al-Sarem, M. Al-Asali, A. Y. Alqutaibi, and F. Saeed, "Enhanced Tooth Region Detection Using Pretrained Deep Learning Models," International Journal of Environmental Research and Public Health, 19, 15414, 2022.

[57] K. L. Du and M. Swamy, "Combining Multiple Learners: Data Fusion and Ensemble Learning," In Neural Networks and Statistical Learning; Springer: Berlin/Heidelberg, Germany, pp. 737–767, 2019.

[58] M. Al-Sarem, F. Saeed, Z. G. Al-Mekhlafi, B.A. Mohammed, T. Al-Hadhrami, M. T. Alshammari, A. Alreshidi, and T .S. Alshammari, "An optimized stacking ensemble model for phishing websites detection," Electronics, 10(11), p.1285, 2021.

[59] B. Krawczyk, L. L. Minku, J. Gama, J. Stefanowski, and M. Wo´zniak, "Ensemble learning for data stream analysis: A survey," Information Fusion, 37, 132–156, 2017.

[60] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1251-1258, 2017.

[61] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv, arXiv:1409.1556, 2014.