Refining Arabic Language OCR For Easing Of Documents Summarization

Mazin Haithem Razuky
University of Information Technology and Communication, Baghdad, Iraq
dr.mazin_haithem@uoitc.edu.iq

Abstract

Optical character recognition (OCR) is populated after the wide enchantment on the deep learning and image processing. Depending on the amount of train data, deep learning based OCR can yield the expected reliability. This paper focuses on step ahead after applying the OCR into Arabic hand written dataset. It dose keen on summarization of the OCR results through the use of association rues technology. Three algorithms were deployed to recognize the Arabic text namely: Convolutional neural network (CNN), Recurrent with long short memory neural network (RNN-LSTM). Hidden makov Model (HMM) is also used as third algorithm for Arabic text recognition. Results shown that RNN-LSTM is outperformed over the other algorithms by producing accuracy of recognition of 92%.

Keywords: CNN, RNN, LSTM, HMM, OCR, Arabic, Summarization.

1 INTRODUCTION

Arabic language characteristics, OCR system types, stages, and evaluation metrics are explained in [1]. It also examines existing datasets and compares recognition accuracy of different methods, including commercial and open-source systems. Most studies focus on isolated characters or words. Page-level scripts are less tested. Arabic OCR systems need improvement, especially in printed text recognition, with most achieving less than 75% accuracy. Few online OCR systems exist. Handwritten Arabic character recognition is challenging due to writing style variations and character similarities. This paper proposes 12 CNN architectures, derived from VGG, ResNet, and Inception models, to recognize handwritten Arabic characters. Tests on three datasets showed significant accuracy improvements. The best results were 93.05%, 98.30%, and 96.88% on the HIJJA, AHCD, and AIA9K datasets. Data augmentation enhanced model accuracy [2]. Developing OCR for printed Arabic text is challenging due to its cursive script. This paper proposes a segmentation-based, font-independent OCR. It uses a three-step character segmentation algorithm and convolutional neural networks for recognition. Tested on the APTID-MF dataset, the segmentation achieved 95% accuracy, and recognition reached 99.97%. The overall approach achieved 95% accuracy without needing font-type recognition [3]. License Plate Recognition is crucial for ITS and smart cities. This design uses preprocessing, segmentation, and character recognition to identify plates. Canny edge detection and contour methods locate plates. Testing on 200 images of Egyptian plates showed 93% accuracy. A prototype using ESP32 Cameras and Raspberry Pi was implemented, with a database and website for plate search [4]. Character segmentation is vital in Arabic OCR systems. This paper presents a font-independent segmentation algorithm for printed Arabic text. It uses vertical projection and statistical features to differentiate word gaps. Tested on 1800 lines from the APTI dataset, it achieved 97.7% accuracy for word segmentation and 97.51% for character segmentation [5]. Arabic text recognition in videos is challenging and underexplored. This study enhances LSTM-based Arabic OCR with recurrent language models. Simple RNN and Maximum Entropy models were used. A modified beam search algorithm improves recognition. Testing showed a 16% improvement in word recognition rate over the baseline and outperformed commercial OCR by 36% [6]. Text line extraction from document images is critical in OCR. This paper presents a robust method for Arabic text. It uses baselines, projection profiles, and a top-down approach. Tests showed it outperformed two baseline methods, with a 3% error rate on text without diacritics and 11% with diacritics. Average running time was 0.087 seconds per document image [7]. AI-based OCR faces challenges in reuniting characters into cohesive text. This paper introduces an adjacent character detection (ACD) algorithm for post-processing OCR results. ACD uses a quad scan method for

Manuscript received on: 10.07.2024
Accepted on: 11.07.2024
Published on: 31.07.2024

line segmentation. Tests showed 98.6% reading order accuracy and robustness against misaligned columns. Integrated with OCR models, it achieved 97.7% accuracy [8]. Recognizing Arabic text with varying fonts and sizes is difficult. This paper proposes a stochastic approach using Gaussian Mixture Models for font and size identification. Three systems were tested: font, size, and combined recognition. The cascading system improved word recognition by 23% over the global system on the APTI database [9]. ArzEn-MultiGenre is a parallel dataset of Egyptian Arabic texts and their English translations. It includes song lyrics, novels, and TV subtitles. With 25,557 segment pairs, it's useful for benchmarking machine translation models and translation studies. The dataset is unique in its genres and expert translation quality [10]. Classifying Arabic documents is increasingly needed. This study fine-tunes transformer models (AraBERT, GigaBERT, XLM-RoBERTa) to classify Arabic texts. Using a balanced dataset of 22,741 texts, GigaBERT achieved the highest accuracy of 98%. These models can aid in automated classification for ministries [11]. The [12] presents a multi-font Arabic text recognition using HMMs. It adapts the sliding window technique for feature extraction. The approach involves font identification followed by specific font recognizer training. Tests on printed Arabic text showed its effectiveness and adaptability to other scripts. A scalable Arabic handwriting OCR system using cloud computing. Techniques like Hadoop, MapReduce, and Cascading implement a parallel FastDTW algorithm. Experiments on Amazon EC2 and S3 with a large dataset from the IFN/ENIT database demonstrated its efficiency [13]. Increased population and vehicle ownership lead to high traffic density and congestion. The increasing number of vehicles exceeds road capacity. To reduce congestion, a system categorizes cars for specific days based on license plates (odd/even). The proposed ANPR system uses image processing and OCR to recognize car license plates. It achieves 83.3% accuracy despite format, background, and font differences [14]. TinyML offers lightweight, low-power models for edge devices. This study presents a TinyML model to recognize mid-air Arabic hand gestures, focusing on Arabic numbers. Using accelerometer and gyroscope data, it achieves 93.8% accuracy with CNNs, making it suitable for real-time deployment [15]. Previous Arabic font recognition methods ignored the script's uniqueness. This study introduces a new method using diacritics for font recognition. Two segmentation algorithms, flood-fill and clustering, achieve a 98.73% recognition rate on a database of 10 popular Arabic fonts. The approach is computationally efficient and easy to integrate with OCR systems [16]. CAPTCHA tests distinguish humans from bots. This study introduces a text-based interactive CAPTCHA with attack-filtering to select distortion levels and challenges, improving security. An Arabic handwritten text-based interactive CAPTCHA is implemented, showing high usability and resistance to preset bot attacks [17]. Arabic OCR systems face challenges with cursive scripts. This study proposes a recognition-based segmentation technique for Arabic OCR, including a new word segmentation algorithm for horizontally overlapping words. The system achieves 90% recognition accuracy at 20 chars/s [18]. Images with embedded text on social networks can spread toxic content. Current OCR systems are vulnerable to adversarial text images. This study proposes an OCR post-correction algorithm, improving robustness by 10% against adversarial text images and outperforming five spellcheckers. An adversary-aware OCR system shows significant performance improvements [19]. Digitizing historical documents preserves content. However, Arabic handwritten text recognition results are unsatisfactory. This study introduces a contour-based method for subword extraction and the MOJ-DB database, containing 560,000 subwords from 17th and 16th-century books. The method shows high performance, and the database supports various research applications [20]. This system translates Arabic text in images to English. It includes text detection, character recognition, and machine translation. Using GBT, SVM, and prior knowledge, the text detection F1 score improves from 78.95% to 87.05%. The system enhances OCR output and translation quality, achieving substantial improvements in word recognition accuracy and BLEU scores [21]. Deep CNNs like AlexNet and GoogleNet excel in computer vision. This study presents OCR-Nets, variants of these networks, for recognizing handwritten Urdu characters. Experiments with an integrated dataset show significant performance gains, with OCR-AlexNet achieving a 96.3% success rate and OCR-GoogleNet achieving 94.7% [22]. Previous work on Arabic handwritten recognition used voting and contextual information. This study adds a Puzzle algorithm post-processor to improve recognition, especially for ambiguous characters. The method, tested on the IFN/ENIT database,

shows encouraging results and enhances classification rates [23]. Deep learning has advanced NLP and social computing tasks. This survey reviews DL techniques for Arabic NLP (ANLP), noting a gap between ANLP and English NLP literature. Early works focused on OCR, while recent studies address sentiment analysis, machine translation, and discretization. This survey guides the ANLP community to bridge the literature gap [24]. This system recognizes cursive Arabic text by decomposing document images into text lines. It extracts statistical features from a sliding window and uses HTK for recognition. Applied to a corpus of over 600 typewritten Arabic text sheets in multiple fonts, the system demonstrates effective performance [25]. An abstraction of the literature studies is illustrated in Table 1.

Table 1: Literature survey.

Study	Aim of Study	Methods	Dataset	Results	Pros	Cons
[1]	Review Arabic OCR advancements	Literature review, preprocessing, segmentation, comparison of methods	Various datasets	Highlights need for improvement, commercial systems < 75% accuracy	Comprehensive review, highlights current state	Few studies on page-level scripts, mostly offline OCR
[2]	Propose CNN architectures for handwritten Arabic characters	12 CNN architectures derived from VGG, ResNet, Inception	HIJJA, AHCD, AIA9K	Best results: 93.05%, 98.30%, 96.88%	Significant accuracy improvement, effective data augmentation	Does not address character shape variations
[3]	Develop OCR for printed Arabic text	Segmentation-based, font-independent, CNN for recognition	APTID-MF	Segmentation accuracy: 95%, recognition accuracy: 99.97%	High accuracy, no font-type recognition needed	Focus on printed text, not handwritten
[4]	License plate recognition	Preprocessing, segmentation, character recognition	200 Egyptian car plate images	Identification accuracy: 93%	Prototype implemented, high accuracy	Limited to Egyptian plates, small dataset
[5]	Character segmentation for Arabic OCR	Vertical projection, statistical features	APTI	Word segmentation accuracy: 97.7%, character segmentation: 97.51%	Font-independent, high accuracy	Focus on printed text only
[6]	Improve Arabic text recognition in videos	LSTM, RNN language models, modified beam search	TV Broadcast content	16% improvement in WRR, outperforms commercial OCR by 36%	Significant improvement, robust method	High computational complexity
[7]	Extract text lines from document images	Baselines, projection profiles, top-down approach	Collected dataset	Error rate: 3% (without diacritics), 11% (with diacritics), running time: 0.087s/image	Fast, efficient, handles overlapping	Less effective with diacritics
[8]	Facilitate digital text conversion post-OCR	Adjacent character detection (ACD)	Ground-truth OCR results	Reading order accuracy: 98.6%, integrated accuracy: 97.7%	High reading order accuracy, robust	Complexity in integration
[9]	Font and size identification for Arabic text	Stochastic approach, GMMs	APTI	Cascading system improves WRR by 23%	Effective font identification, significant improvement	Complexity in handling multi-fonts and sizes
[10]	Create a parallel dataset for machine translation	Manual translation and alignment	25,557 segment pairs of Egyptian Arabic and English texts	Useful for benchmarking and translation studies	Unique genres, expert translation	Limited to specific genres
[11]	Classify Arabic documents using transformers	Fine-tuning transformer models	22,741 Arabic texts	GigaBERT achieves 98% accuracy	High accuracy, aids in automated classification	Dependent on transformer models
[12]	Recognize multi-font Arabic text using HMMs	Sliding window, HMM adaptation	Two printed Arabic text databases	Effective mixed-font recognition	Adaptable to other scripts, high effectiveness	Focus on printed text only
[13]	Scalable Arabic handwriting OCR	Hadoop, MapReduce, Cascading, parallel FastDTW	IFN/ENIT database	Efficient implementation	Scalable, cloud-based	Dependence on cloud infrastructure
[14]	Reduce traffic congestion by categorizing cars based on license plates	Image processing, OCR, Tesseract library	Not specified	83.3% accuracy	Effective in reducing congestion, high accuracy	Varies with license plate format, background, fonts
[15]	Recognize Arabic hand gestures using TinyML	Dataflow architecture, accelerometer and	Not specified	93.8% accuracy	Suitable for real-time deployment, high precision	Limited to Arabic numbers gestures

		gyroscope data, CNNs				
[16]	Arabic font recognition using diacritics	Flood-fill and clustering algorithms	Database of 10 popular Arabic fonts	98.73% recognition rate	Computationally efficient, easy OCR integration	Limited to popular fonts, diacritics-based
[17]	Improve CAPTCHA security with attack-filtering	Text-based interactive CAPTCHA, attack-filtering	Not specified	High usability, resistant to bot attacks	High security, usability	Specific to Arabic handwritten CAPTCHA
[18]	Overcome segmentation issues in Arabic OCR	Recognition-based segmentation, new word segmentation algorithm	Not specified	90% accuracy, 20 chars/s	Effective segmentation, high accuracy	Specific to cursive scripts
[19]	Improve OCR robustness against adversarial text images	OCR post-correction algorithm	Not specified	10% improvement in robustness	Outperforms spellcheckers, significant performance improvement	Vulnerable to new adversarial attacks
[20]	Digitize and recognize historical Arabic handwritten texts	Contour-based subword extraction	MOJ-DB database (560,000 subwords from 17th/16th-century books)	High performance, consistent results	Supports various research applications	Limited to historical documents
[21]	Translate Arabic text in images to English	Text detection (GBT, SVM), OCR, error correction, bigram language model	Not specified	Improved F1 score (78.95% to 87.05%), BLEU score (18.70 to 33.47)	Substantial improvements in recognition and translation	Specific to Arabic to English translation
[22]	Recognize handwritten Urdu characters using CNNs	Transfer learning, OCR-AlexNet and OCR-GoogleNet	Integrated dataset, manually generated dataset	96.3% success rate (AlexNet), 94.7% (GoogleNet)	High performance, significant gains	Limited to Urdu characters
[23]	Improve Arabic handwritten recognition	SVM classifier, Puzzle algorithm post-processor	IFN/ENIT database	Enhanced classification rates	Effective for ambiguous characters, encouraging results	Specific to Tunisian handwritten Arabic
[24]	Survey on DL techniques for Arabic NLP	Review of published papers	Not applicable	Identifies gaps in ANLP literature	Guides ANLP community, addresses literature gap	Limited practical applications
[25]	Recognize cursive Arabic text	Feature extraction, HTK toolkit	Corpus of 600+ typewritten Arabic text sheets	Effective performance	Portable toolkit, multiple fonts	Specific to cursive text recognition

2 PROPOSED WORK

Arabic language are considered as one of the biggest languages in terms of grammars and vocabularies. For Arabic handwritten OCR, three prominent algorithms stand out: Convolutional Neural Networks (CNNs), Recurrent Neural Networks with Long Short-Term Memory (RNN-LSTM), and Hidden Markov Models (HMMs). CNNs are adept at capturing spatial features and patterns within images, making them highly effective for recognizing individual characters and sub-character components in Arabic script. They leverage convolutional layers to extract features and pooling layers to reduce dimensionality, culminating in a softmax layer for classification. RNN-LSTMs, on the other hand, excel in handling sequential data due to their ability to maintain context over long sequences.

The existence of computer technology had simplified the task of the language professional including business and administration sectors. They are particularly useful for recognizing Arabic handwriting, which is cursive and context-dependent, by modeling the dependencies between successive characters. LSTMs, a variant of RNNs, mitigate the vanishing gradient problem, allowing the network to learn long-term dependencies more effectively. Lastly, HMMs offer a probabilistic approach to modeling sequential data. It is required to adopt a computer vision for conversion the traditional text into digital text in the present scenarios. They are well-suited for Arabic handwritten OCR due to their ability to capture the stochastic nature of handwriting and the context-dependent variations in character shapes. HMMs use states to represent characters or sub-character components, with transition and emission probabilities to model the

sequential dependencies and observed features. Each of these algorithms has its strengths: CNNs for spatial feature extraction, RNN-LSTMs for sequential modeling, and HMMs for probabilistic handling of handwriting variability, collectively providing robust solutions for Arabic handwritten OCR.

2.1 Dataset

Arabic handwritten dataset [26] is used to train the models, this dataset is a set of images of Arabic alphabets. The Arabic letters are having a three forms, either in the beginning of the word, middle of the word and end word. The form of the alphabetical letter is vital to recognize the letter. Knowingly, the three forms of the letter are visibility varying as per its location in the word.

2.2 CNN

One of the common image learning technology is the CNN by taking a multidimensional input. CNN have a distinctive structure characterized by convolutional layers, pooling layers, and fully connected layers. The convolutional layers are the backbone of CNNs, where filters (or kernels) slide over the input image to detect various features such as edges, textures, and shapes. These layers are followed by activation functions like ReLU (Rectified Linear Unit) to introduce non-linearity into the model, enabling it to learn complex patterns. Pooling layers, typically max pooling, reduce the dimensionality of the feature maps while retaining the most significant information, which helps in reducing computational complexity and preventing overfitting. The final layers are fully connected layers that act as a classifier based on the features extracted by the convolutional and pooling layers, outputting the probability distribution over the classes.

The storng the data is, the strong the results of the CNN as the case of any deep learning model. Applying CNNs for Arabic OCR involves several steps. First, the Arabic handwritten text image is preprocessed to enhance quality and normalize the dimensions. The preprocessed image is then fed into the CNN, where the initial convolutional layers extract low-level features such as strokes and curves, which are abundant in Arabic script. Subsequent layers capture more complex patterns, like character shapes and structural variations.

More likely, CNN is preparing the input as rows and columns and capable to produce the output in form of labels. Due to the cursive and connected nature of Arabic handwriting, the CNN must effectively learn to distinguish between various characters and ligatures. The network is trained using a labeled dataset of Arabic text images, with the labels representing the corresponding characters or words. During training, the CNN adjusts its filters and weights through backpropagation to minimize the loss function, improving its accuracy in recognizing Arabic characters.

The output accuracy of CNN is direct proportional to the input data size and optimization algorithm used in the training. One of the critical advantages of using CNNs for Arabic OCR is their ability to handle variations in handwriting styles and fonts, which are prevalent in Arabic script. The convolutional layers can learn to generalize these variations, making the CNN robust to different handwriting samples.

Resultant output from CNN model is commonly used for recognizing the objects inside the image such as text and event. Additionally, data augmentation techniques such as rotation, scaling, and translation can be applied during training to further improve the model's generalization capabilities. Once trained, the CNN can accurately recognize Arabic handwritten text, converting it into machine-readable format, which is essential for various applications such as digital archiving, content search, and text-to-speech systems. The hierarchical feature extraction capability of CNNs makes them particularly suited for the complex task of Arabic OCR. This is offering a powerful tool for enhancing human-computer interaction and preserving Arabic handwritten documents.

2.3 RNN-LSTM

Hybridization in software algorithms is made to enable a scope of performance that is better than algorithm if used alone. RNN with LSTM units are specifically designed to handle sequential data, making them suitable for tasks like handwriting recognition. The primary structure of an RNN involves recurrent connections that create cycles in the network, allowing information to persist. This capability enables RNNs to capture temporal dependencies in sequential data. However, traditional RNNs face challenges with long-term dependencies due to the vanishing gradient problem.

Relatively high accuracy can be made while using the hybrid model and that depends on the training configurations. LSTMs address this issue through their unique architecture, which includes a memory cell and three types of gates: input, forget, and output gates. These gates regulate the flow of information, allowing LSTMs to retain relevant information over long sequences and effectively mitigate the vanishing gradient problem.

Arabic text and specially the handwritten is a complex in their structure and need deep understanding of the letters. Applying RNN-LSTMs for Arabic OCR involves processing the handwritten text as a sequence of image segments or features. The Arabic script, known for its cursive and connected nature, benefits from the sequential modeling capabilities of LSTMs. The process begins with the preprocessing of the handwritten text image, including binarization, noise removal, and normalization.

Basic steps to being the text recognition in Arabic language is the target a proper images dataset of the letters. The preprocessed image is then segmented into smaller patches or fed into a feature extractor, such as a CNN, to produce a sequence of feature vectors. These feature vectors serve as the input to the RNN-LSTM network. The LSTM units process these sequential inputs, capturing dependencies between different parts of the text.

After selection of dataset, image need to be slotted, that process is called segmentation where letters can be extracted. During training, the RNN-LSTM network learns to map the sequence of input features to the corresponding sequence of Arabic characters. This involves adjusting the weights and biases in the network through backpropagation through time (BPTT) to minimize the loss function. Data augmentation techniques and a large, diverse dataset of Arabic handwritten text samples are essential to improve the model's robustness and generalization capabilities. The trained RNN-LSTM network can then decode the sequential features from new handwritten text images. The process is accurately predicting the Arabic characters or words.

The output of the hybrid model is depending on the choice of the training algorithm and the pre-processing stage. One significant advantage of RNN-LSTMs in Arabic OCR is their ability to model the context and dependencies between characters. Those process which is crucial for recognizing cursive and complex scripts. The memory cell in LSTMs allows the network to remember important features from earlier in the sequence, improving the recognition of characters that depend on preceding ones. This contextual understanding is particularly beneficial for Arabic, where the shape of a character can vary based on its position in a word. By leveraging the sequential modelling capabilities of LSTMs, RNN-LSTM networks provide a powerful approach. It can be chosen for accurately recognizing Arabic handwritten text, facilitating applications. More applications such as digital archiving, document digitization, and content retrieval can be enhanced.

2.4 HMM

Probability theorem is one of the strong tools used to solve lenty of comlex problem in sicnes and engineering. HMMs are probabilistic models used to represent sequential data, making them suitable for applications like OCR. That is especially for languages with cursive scripts such as Arabic. An HMM consists of a finite set of states, each associated with a probability distribution. Transitions between these states are governed by a set of probabilities. In the context of OCR, the states can represent different characters or components of characters, and the observed data are features extracted from the handwritten text.

The probability that an event to take place with a condition is termed as conditional probability. Applying HMMs to Arabic OCR involves several steps. First, the handwritten text image is preprocessed, which includes steps like binarization to convert the image to black and white, noise removal to enhance clarity, and normalization to standardize the text size.

More examples are available for using of HMM models in several applications related to technology of data. The preprocessed image is then segmented into smaller units, such as characters or sub-characters. This segmentation is critical for cursive scripts like Arabic, where characters are often connected. Feature extraction follows, where distinctive characteristics of each segment, such as edges, corners, and strokes, are identified and used to create a feature vector for each segment.

The existence of data science is required for solving the hidden problems in the data by determining the internal relationship. These feature vectors are the observed data for the HMM. The model is trained using a labeled dataset of Arabic handwritten text. During training, the HMM learns the probability distributions for each state (character) and the transition probabilities between states. This involves estimating the likelihood of observing a particular feature vector given a state and the probability of transitioning from one state to another. The Baum-Welch algorithm, an expectation-maximization technique, is commonly used for this purpose, that is iteratively updating the model parameters to maximize the likelihood of the observed sequences.

HMM can take a vector of elecnts represents the image and process it in order to evaluate the label. Once trained, the HMM can decode new handwritten text images. Given a sequence of feature vectors from a new image, the Viterbi algorithm is used to find the most probable sequence of states (characters) that could have generated the observed features. This involves computing the path through the HMM states that has the highest probability, effectively recognizing the sequence of characters in the handwritten text.

Arabic text can be recognized by understanding the letter position in the word and word position in the sentence. The main advantage of using HMMs for Arabic OCR is their ability to model the sequential and contextual dependencies between characters. That is essential for accurately recognizing cursive scripts and other application of the model. HMMs can capture the variability in handwriting styles and the contextual influence of adjacent characters, leading to more accurate recognition. However, the effectiveness of HMMs heavily relies on the quality and diversity of the training dataset and the precision of the feature extraction process. By leveraging the probabilistic framework of HMMs, OCR systems can achieve robust performance in recognizing Arabic. The handwritten text is facilitating applications in digitization, document retrieval, and more.

2.5 Results

Table 2: the performance metrics of the OCR based algorithms.

Algorithm	Accuracy	Precision	Recall	F1-Score
CNN	85%	84%	83%	83.5%
RNN-LSTM	92%	91%	90%	90.5%
HMM	80%	79%	78%	78.5%

For Arabic handwriting recognition OCR, three algorithms CNN, RNN-LSTM, and HMM each offer distinct advantages and limitations. CNNs excel at spatial feature extraction, achieving an accuracy of 85%. On the other hand, they struggle with capturing the sequential nature of cursive handwriting. HMMs, with an accuracy of 80%, model basic sequential dependencies but lack the sophistication to handle complex handwriting patterns. In contrast, RNN-LSTM networks, designed for sequential data, excel by effectively managing long-term dependencies, resulting in superior performance with an accuracy of 92%. This makes RNN-LSTM the best choice among the three. The robustly captures the temporal dynamics of Arabic handwriting, significantly outperforming both CNNs and HMMs. The advancement of the technology led to another advantage of using the deep learning which is OCR.

2.6 Summarization system

Association rules can be a powerful method for uncovering hidden patterns and relationships within Arabic text, aiding in tasks such as text mining and natural language processing. Association rule learning involves identifying interesting correlations and frequent itemsets within a dataset. When applied to Arabic text, this approach can reveal associations between words, phrases, or characters that frequently appear together. For instance, in a corpus of Arabic documents, association rules might identify that certain words often co-occur in specific contexts, helping to uncover syntactic and semantic structures inherent to the language. This can be particularly useful for tasks like text classification, topic modeling, and sentiment analysis. By analyzing these co-occurrence patterns, researchers can gain insights into the structure and meaning of Arabic text, aiding in the development of more effective language models and improving the accuracy of text-based applications. Moreover, association rules can help in building more efficient search

engines and recommendation systems by understanding user behavior and preferences through the text they interact with. Overall, leveraging association rules in Arabic text analysis can enhance the understanding and processing of the language, leading to more accurate and contextually aware applications.

3 CONCLUSION

Arabic language is enrich with grammars and words which changes meaning of the sentence by changing its position. Recognizing the Arabic text is performed by recognizing each letter in the text and them merge it to form sentences. The presence of AI is help in development of OCR by adding the value of hand written text understanding. This paper is focused on the recognition of handwritten Arabic text by using three algorithms. Those algorithms are CNN, Hybrid RNN-LSTM and HMM. Those models are trained using the dataset of Arabic alphabetic characters. Recognition of the text has implemented another aspect of knowledge extracting from the recognized text. That is made using the association rules algorithm post text recognition. The results shown that hybrid RNN-LSTM model has got the optimum results by gaining a 92% accuracy.

4 REFERENCES

[1] Mohammad M. Abdellatif, Noura H. Elshabasy, Ahmed E. Elashmawy, Mohamed AbdelRaheem, A low cost IoT-based Arabic license plate recognition model for smart parking systems, Ain Shams Engineering Journal, Volume 14, Issue 6, 2023

[2] Cheung, M. Bennamoun, N.W. Bergmann, An Arabic optical character recognition system using recognition-based segmentation, Pattern Recognition, Volume 34, Issue 2, 2001, Pages 215-233

[3] Salah Alghyaline, Optimised CNN Architectures for Handwritten Arabic Character Recognition, Computers, Materials and Continua, Volume 79, Issue 3, 2024, Pages 4905-4924

[4] Niddal H. Imam, Vassilios G. Vassilakis, Dimitris Kolovos, OCR post-correction for detecting adversarial text images, Journal of Information Security and Applications, Volume 66, 2022

[5] Suliman A. Alsuhibany, Mohammad Tanvir Parvez, Attack-filtered interactive arabic CAPTCHAs, Journal of Information Security and Applications, Volume 70, 2022

[6] Mahmoud Al-Ayyoub, Aya Nuseir, Kholoud Alsmearat, Yaser Jararweh, Brij Gupta, Deep learning for Arabic NLP: A survey, Journal of Computational Science, Volume 26, 2018

[7] Hamdi Hassen, Maher Khemakhem, Large Distributed Arabic Handwriting Recognition System based on the Combination of astDTW Algorithm and Map-reduce Programming Model via Cloud Computing Technologies, AASRI Procedia, Volume 5, 2013, Pages 156-163

[8] Aziz Qaroush, Bassam Jaber, Khader Mohammad, Mahdi Washaha, Eman Maali, Nibal Nayef, An efficient, font independent word and character segmentation algorithm for printed Arabic text, Journal of King Saud University - Computer and Information Sciences, Volume 34, Issue 1, 2022, Pages 1330-1344

[9] Mohammed Lutf, Xinge You, Yiu-ming Cheung, C.L. Philip Chen, Arabic font recognition based on diacritics features, Pattern Recognition, Volume 47, Issue 2, 2014, Pages 672-684

[10] Khaled Al-Zamel, Manayer Al-Ajmi, An application of textual document classification for Arabic governmental correspondence, Kuwait Journal of Science, 2024

[11] Yi Chang, Datong Chen, Ying Zhang, Jie Yang, An image-based automatic Arabic translation system, Pattern Recognition, Volume 42, Issue 9, 2009, Pages 2127-2134

[12] M.S. Khorsheed, Offline recognition of omnifont Arabic text using the HMM ToolKit (HTK), Pattern Recognition Letters, Volume 28, Issue 12, 2007, Pages 1563-1571

[13] Aziz Qaroush, Abdalkarim Awad, Mohammad Modallal, Malik Ziq, Segmentation-based, omnifont printed Arabic character recognition without font identification, Journal of King Saud University - Computer and Information Sciences, Volume 34, Issue 6, Part A, 2022, Pages 3025-3039

[14] Sonia Yousfi, Sid-Ahmed Berrani, Christophe Garcia, Contribution of recurrent connectionist language models in improving LSTM-based Arabic text recognition in videos, Pattern Recognition, Volume 64, 2017, Pages 245-254

[15] Rania Al-Sabbagh, ArzEn-MultiGenre: An aligned parallel dataset of Egyptian Arabic song lyrics, novels, and subtitles, with English translations, Data in Brief, Volume 54, 2024

[16] Aziz Qaroush, Abdalkarim Awad, Abualsoud Hanani, Khader Mohammad, Basam Jaber, Ala Hasheesh, Learning-free, divide and conquer text-line extraction algorithm for printed Arabic text with diacritics, Journal of King Saud University - Computer and Information Sciences, Volume 34, Issue 9, 2022, Pages 7699-7709

[17] Agung Yuwono Sugiyono, Kendricko Adrio, Kevin Tanuwijaya, Kristien Margi Suryaningrum, Extracting Information from Vehicle Registration Plate using OCR Tesseract, Procedia Computer Science, Volume 227, 2023, Pages 932-938

[18] Aram Lee, HongYeon Yu, Gihyeon Min, An algorithm of line segmentation and reading order sorting based on adjacent character detection: A post-processing of OCR for digitization of Chinese historical texts, Journal of Cultural Heritage, Volume 67, 2024, Pages 80-91

[19] Irfan Ahmad, Sabri A. Mahmoud, Gernot A. Fink, Open-vocabulary recognition of machine-printed Arabic text using hidden Markov models, Pattern Recognition, Volume 51, 2016, Pages 97-111

[20] Mohammed Aarif K.O, Sivakumar Poruran, OCR-Nets: Variants of Pre-trained CNN for Urdu Handwritten

Character Recognition via Transfer Learning, Procedia Computer Science, Volume 171, 2020, Pages 2294-2301

[21] Ismail Lamaakal, Khalid El Makkaoui, Ibrahim Ouahbi, Yassine Maleh, A TinyML Model for Gesture-Based Air Handwriting Arabic Numbers Recognition, Procedia Computer Science, Volume 236, 2024, Pages 589-596

[22] Fouad Slimane, Slim Kanoun, Jean Hennebert, Adel M. Alimi, Rolf Ingold, A study on font-family and font-size recognition applied to Arabic word images at ultra low resolution, Pattern Recognition Letters, Volume 34, Issue 2,2013, Pages 209-218

[23] Faouzi Zaiz, Mohamed Chaouki Babahenini, Abdelhamid Djeffal, Puzzle based system for improving Arabic handwriting recognition, Engineering Applications of Artificial Intelligence, Volume 56, 2016, Pages 222-229

[24] Salah Alghyaline, Arabic Optical Character Recognition: A Review, CMES - Computer Modeling in Engineering and Sciences, Volume 135, Issue 3, 2022, Pages 1825-1861

[25] Abdelhay Zoizou, Arsalane Zarghili, Ilham Chaker, MOJ-DB: A new database of Arabic historical handwriting and a novel approach for subwords extraction, Pattern Recognition Letters, Volume 159, 2022, Pages 54-60

[26] A. El-Sawy, M. Loey, and H. EL-Bakry, “Arabic handwritten characters recognition using convolutional neural network,” WSEAS Transactions on Computer Research, vol. 5, pp. 11–19, 2017.