Journal of Theoretical and Applied Information Technology
15th March 2024. Vol.102. No 5

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

HYBRID FEATURE-DRIVEN ENSEMBLE LEARNING IN ARABIC NLP: FUSING SEQUENTIAL NEURAL NETWORKS WITH ADVANCED TEXT ANALYSIS TECHNIQUES

EMAN ALJOHANI
Department of Computer Science, College of Computer Science and Engineering, Taibah University,
Madinah 42353, Saudi Arabia
E-mail: Emmjohani@taibahu.edu.sa

ABSTRACT

This paper investigates the application of deep learning methodologies for Arabic news classification, with a primary focus on the role of text preprocessing in enhancing model performance. By evaluating stemming across a range of datasets, the study hopes to clarify how effective stemming is at enhancing classification results. This study provides an extensive comparative analysis of deep learning models' effects on Arabic text processing. We introduce a novel hybrid neural network that combines TF-IDF weighting and FastText embeddings with Sequential Networks layers for NLP text classification. This architecture uses both static and dynamic language features for improved classification, capturing both the temporal word dependencies and the granular semantics. This paper proposed the new Hybrid-Bi model, a sophisticated approach that combines hybrid feature, BiLSTM, BiGRU, SVM, XGBoost, and Random Forest using stacking. It consistently performs better than other approaches on a range of news sources with and without stemming. The data also reveals a tendency where models generally outperform stemmed models, suggesting that stemming may omit important semantic information necessary for precise interpretation and classification in Arabic NLP. It achieves peak accuracies of up to 0.98 in Arabiya and Khaleej, especially in no stemming scenarios.
Keywords: *Text Classification, NLP, Feature Extraction, Deep Learning, Ensemble Learning*

1. INTRODUCTION

Arabic news classification, a critical component of natural language processing, is undergoing a radical transformation thanks to the integration of deep learning techniques. In the constantly evolving digital world, providing users with timely, relevant, and personalized information requires effective categorization of Arabic news articles. This paper explores the application of deep learning techniques to address the complexities of Arabic text, with the goal of increasing the accuracy and efficacy of news classification. We examine the challenges posed by dialectal variations, linguistic nuances, and the dynamic nature of news content, as well as the complexities of Arabic language processing. A detailed analysis of various deep learning models, including fastText and Glove, and the development of hybrid ensembles that combine traditional and innovative methods are used to outline the journey.

Arabic news texts pose particular difficulties because of the language's linguistic complexity and wide cultural range. The difficulties include the complexity of Arabic words' morphology, the existence of dialectal variances, and diacritical markings (Tashkeel). These elements add to the intrinsic complexity of interpreting Arabic news text, necessitating the use of specialized techniques to get past linguistic obstacles and correctly classify material (Khalifa, Zalmout and Habash, 2020). It is crucial to comprehend and tackle these obstacles in order to create strong and efficient Arabic news classification models. This paper explores the unique challenges encountered when working with Arabic news text, offering insights into the nuances that require customized solutions.

The main objectives of this study are summarized as follows:

- The study conducts a thorough and in-depth examination of the Arabic text classification workflow procedure, which encompasses preprocessing, feature extraction, and classification.
- Assesses the impact of incorporating stemming in Arabic text classification. This evaluation investigates how stemming affects the performance of the classification workflow,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

shedding light on its significance in Arabic text classification tasks.

- The study leverages pretrained transfer models, which are pre-existing machine learning models trained on large datasets.
- This work presents a hybrid machine learning feature that combines embeddings and TF-IDF weighted features to improve natural language processing.
- To increase Arabic text classification accuracy, we proposed an integrated approach that combines sophisticated our hybrid feature with ensemble learning techniques.

These combined contributions demonstrate a comprehensive exploration of the text classification process for Arabic text, encompassing the utilization of state-of-the-art techniques, ensemble learning, and an in-depth analysis of the influence of stemming on classification results.

The paper is arranged as follows after the introduction, where we define the main topic and outline our goals. Section 2 explores the Related Works and provides an overview of the learning models that have been previously studied in this area. A thorough explanation of our methodology is given in Section 3, "The Proposed Model," which includes subsections 3.1 on dataset selection, 3.2 on data preprocessing techniques, 3.3 on feature extraction methodology, 3.4 on base classifiers, 3.5 on deep learning models, and 3.6 on our ensemble learning strategy. We evaluate the impact of base classifiers (4.1), hybrid models with different word embeddings (4.2), and ensemble learning (4.3) on the overall performance in Section 4, "Results and Analysis."

2. RELATED WORKS

This section highlights related research on ensemble learning applied to text classification using baseline classifiers from both machine learning and deep learning. In the context of Arabic text classification, seven datasets have emerged, which were employed in the investigation conducted by (Ababneh, 2022). These datasets include the Single-Label Arabic News Articles Dataset (SANAD), Khaleej, Arabiya, Akhbarona, KALIMAT, Waten2004, and Khaleej2004. The investigation aims to determine which of these datasets can offer effective training and impartial evaluation for text classification tasks. Various machine learning models, such as naive Bayes, random forest, K-nearest neighbor, support vector machines, and logistic regression, are employed in this research. (Elnagar, Einea and Al-Debsi, 2019) contributed to the Arabic computational linguistics research community by describing a new large corpus for single-label Arabic text categorization tasks. SANAD, comprising three datasets gathered from annotated Arabic news articles, exhibited balanced characteristics in the Khaleej dataset, while the Akhbarona and Arabiya datasets were imbalanced. Test findings demonstrated robust performance on the SANAD corpus, achieving a maximum accuracy of 95.81%. Recent research papers with a primary focus on Arabic text classification have been increasingly emerging. Two sizable datasets were introduced by (Rifai, Al Qadi and Elnagar, 2021) and were compiled from a variety of Arabic news portals. The first dataset consists of 90,000 single-labeled articles spanning four domains (Business, Middle East, Technology, and Sports), while the second dataset comprises over 290,000 multi-tagged articles. For the evaluation of the single-label dataset, a diverse set of ten shallow learning classifiers was employed. Furthermore, an ensemble model that incorporates the majority-voting technique of all the studied classifiers was introduced. The research employed classifiers like XGBoost and Logistic Regression that were suitable for multi-labeling tasks.

On the other hand, (Qadi et al., 2019) presented a newly constructed dataset containing nearly 90,000 Arabic news articles, complete with their associated tags sourced from Arabic news portals. This dataset was categorized into four main sections: Business, Sports, Technology, and the Middle East. The research employed a classical supervised machine learning classifier in its analysis. In Arabic Text Categorization, a Moroccan News Articles Corpus was presented by (Jbene et al., 2021), comprising data collected from four major Moroccan news websites. To demonstrate the dataset's utility, an evaluation was conducted within the framework of text classification, employing four distinct Machine Learning baselines: Random Forest (RF), Multinomial Naive Bayes (MNB), Support Vector Machine (SVC), and Gradient Boosting (GradBoost) Classifiers.

A framework for sentiment analysis developed by (Minaee, Azimi and Abdolrashidi, 2019) is based on a combination of LSTM and CNN models. In this study, word embeddings for each term within the reviews were obtained through Glove embeddings. These embeddings were subsequently input into both the CNN and LSTM models to generate predictions. The final predictions were computed by averaging the predicted scores from both the LSTM and CNN models. The authors of (Bani Younes, Bani Younes and Al-khdour, 2022) offer a model that categorizes 26 different dialects of Arabic from a given text.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Three classifiers make up the proposed model, which then ensembles their predictions to predict the outcome. Author employs the OneVsAll tactic the main concept behind this technique is to reduce a multi-class classification issue to a binary classification issue. Additionally, the author used both probability predictions and n-gram features. (Rifai, Al Qadi and Elnagar, 2021) has developed a multi-label classification system for Arabic articles and introduced a multi-labeled dataset comprising 293,000 Arabic articles, along with their associated tags, collected from 10 news portals. In their research, they conducted experiments involving the combination of both CNN and RNN architectures, alongside the use of (Conditional Random Rields) CRF-CNN. These novel approaches yielded superior results when compared to traditional machine learning models. By combining Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) networks, without using CNN's Max-pooling layer, the authors of this study present a novel method for Arabic sentiment analysis (Alayba and Palade, 2022). Linguistic nuance representation is improved through the integration of Arabic word normalization tools. Using the same corpora, several Arabic word embedding models are investigated to see how they affect sentiment classification. The authors showcase a novel approach to Arabic sentiment classification that achieves cutting edge outcomes, demonstrating how well their creative methodology captures subtle aspects of Arabic text.

(Elghannam, 2021) introduced a methodology for constructing features in the field of text classification. Their approach revolves around creating a unique bigram alphabet and employing it to represent the content of documents by utilizing term frequency as a weighting mechanism. Through the implementation of an SVM-SMO classifier, their approach demonstrated remarkable proficiency in effectively classifying collections of both Arabic and English text documents. (Zamzami, Himdi and Sabbeh, 2023) explore hybrid network structures in this work, combining CNN with GRU, CNN with BiLSTM, and CNN with LSTM. CNN was selected due to its exceptional performance. Results of applying hybrid models, which are renowned for their ability to extract features, to the Hajj dataset are given. Interestingly, the word vectors in pre-trained embedding were created using Arabic Wordvec and FastText, whereas the word vectors in training the network were learned.

To enhance classification accuracy, some researchers extend the representation by incorporating additional information, such as word n-grams and semantic knowledge. Word n-grams can be used either as standalone feature terms or in conjunction with unigrams (single words). It implies that, when compared to the Bag of Words (BOW) baseline, the classification accuracy is reduced when using only phrases as feature terms. The traditional Bag of Words (BoWs) for data representation does not account for the semantic relationships between words, such as synonymy and hypernymy, when classifying Arabic text (Hijazi, Zeki and Ismail, 2022).

Whereas, (Elghannam, 2021) explored both word-level unigrams and bigrams, compared the classification outcomes with the conventional Bag of Words (BOW) method, and found that the use of N-grams yields superior accuracy in feature representation for classification when contrasted with individual words. A hybrid deep learning approach, combining RNN and CNN architectures, was used for the task of Arabic text categorization. Both static, dynamic, and fine-tuned word embeddings were employed in the process. The experimental results conducted on the Open Source Arabic Corpora (OSAC) dataset demonstrated the effectiveness and high performance of the models proposed by (Ameur, Belkebir and Guessoum, 2020). With a focus on Twitter data, the authors (Ahanin *et al.*, 2023) present two models for classifying emotions in English text. Emoji and mood features are incorporated into the first model, which integrates Word2Vec embeddings, human-engineered features, and deep learning via Bi-LSTM. The second uses Bi-LSTM processing after contextual understanding using the BERT model. These methods represent a novel combination of deep learning features and human-engineered features for efficient emotion analysis. (Berrimi *et al.*, 2023) presents a deep learning model for Arabic sentiment analysis. Bidirectional long short-term memory (BiLSTM) and bidirectional gated recurrent unit (BiGRU) are combined in the model along with an additive attention mechanism. Because of its design, the BiGRU/BiLSTM can process sentences in both directions and successfully pick up on contextual cues. Remarkably, the model uses locally learnable embeddings in addition to FastText embeddings during training.

Our study stands out in the Arabic text classification landscape because it carefully examines the classification workflow and takes a comprehensive approach to feature extraction, preprocessing, and classification. Although previous research has focused on each of these components separately, our work combines them

Journal of Theoretical and Applied Information Technology
15th March 2024. Vol.102. No 5

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

into a cohesive process to provide a more nuanced understanding. We also explore the effect of stemming on Arabic text classification—an aspect often overlooked in the literature. Our analysis fills in a significant research gap while also highlighting the importance of incorporating stemming. Our study also investigates the unexplored possibilities of pretrained transfer models in the Arabic language environment. Using these pre-trained models on large-scale datasets opens up new insights and helps close the current knowledge gap. Moreover, our work presents a novel approach to improve natural language processing in Arabic text: a hybrid machine learning feature that combines TF-IDF weighted features and embeddings. Lastly, we propose an integrated strategy that combines ensemble learning methods with our advanced hybrid feature to improve classification accuracy. These contributions distinguish our study and highlight its role in advancing the understanding and methodologies of Arabic text classification.

3. THE PROPOSED MODEL

We outline the architecture of our innovative system for Arabic text classification. We start by presenting the datasets that serve as the foundation for the training and validation of our model in Section 3.1. Our adopted data preprocessing methods, which are critical to guaranteeing data consistency and quality, are described in Section 3.2. After that, in Section 3.3, the feature extraction procedure is explained. This process is essential for identifying and obtaining significant patterns within the text. Section 3.4 presents the base classifiers, the cornerstone algorithms upon which our model is built. In Section 3.5, we move the emphasis to our cutting-edge deep learning models, emphasizing hybrid models that make use of various word embeddings to accurately represent the subtleties of Arabic. In conclusion, Section 3.6 describes our ensemble learning approach, which combines multiple models to produce better classification accuracy.

3.1 Datasets

SANAD (Single-Label Arabic News Articles Dataset) is a comprehensive collection of Arabic news articles designed for use in various natural language processing (NLP) tasks, such as Text Classification and Word Embedding (Einea, Elnagar and Al Debsi, 2019). The dataset encompasses articles gathered from three prominent Arabic news websites: AlKhaleej, AlArabiya, and Akhbarona. The dataset is notably diverse, featuring seven categories; Culture, Finance, Medical, Politics, Religion, Sports, and Tech, with the exception of AlArabiya, which excludes the Religion category. In total, SANAD comprises over 190,000 articles.

3.2 Data Pre-processing

Normalization help to remove inconsistencies and errors in text data, making it more suitable for downstream natural language processing tasks. Arabic text can use different characters or ligatures to represent the same sounds. Standardization aids in bringing these disparities to a common representation. For instance, normalizing "ة" and "ه" to "ه" guarantees consistency even though they both represent the sound "h" in different contexts. Deleting all non-Arabic characters, numbers, punctuation, special symbols, diacritical marks, and single Arabic letters (letters that do not belong in words). Eliminating stopwords, or words that are overused and unimportant in the text, includes repositions and pronouns are two examples. Diacritical marking removal, or Tashkeel in Arabic, is the process of getting rid of different accents or markings that help you pronounce letters within words (Darwish, Mubarak and Abdelali, 2017).

Tashkeel includes the following: Fatha (فَتْحَة), Damma (ضَمَّة), Kasra (كَسْرَة), Sukun (سُكُون), Shadda (شَدَّة), Tanween Fath (تَنْوِين فَتْح), Tanween Damm (تَنْوِين ضَم), Tanween Kasr (تَنْوِين كَسْر), Maddah (مَدَّة) and Hamzah (هَمْزَة). Preprocessing is highly advantageous as it reduces the size of the index, enhances precision, and harmonizes the classification tasks. The following preprocessing techniques were applied:

- Replaces the Arabic characters (إ, أ, ٱ, آ, ا) with the standard Arabic character " ا ". This helps ensure consistency in text representation.
- Replaces the Arabic character "ى" with "ي"
- Replaces the Arabic character "ؤ" with "ء"
- Replaces the Arabic character "ئ" with "ء"
- Replaces the Arabic character "ة" with "ه"
- Replaces the Arabic character "گ" with "ك"
- To remove repeated characters(Tatweel). It captures any character that occurs more than once and replaces it with the character repeated twice. This is commonly done to handle elongations in Arabic text.
- Normalize hamza as different forms of the Arabic letter "hamza"
- Remove diacritical marks (Tashkeel) from the Arabic text.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

This step's objective is to apply different text preprocessing and normalization techniques to Arabic text in order to improve its consistency, standardization, and suitability for additional analysis, like text mining or natural language processing (NLP). Preprocessing Arabic text is a crucial step in our research to improve text analysis performance (Zerrouki, 2023). To normalize and segment the Arabic text data, we used Tashaphyne, an Arabic light stemmer and segmentor. Reducing words to their base or root form is known as stemming in natural language processing (NLP), and it can be useful for information retrieval, text analysis, and text classification. By reducing word form variations to their common form, stemming attempts to increase text processing and analysis efficiency.

3.3 Feature extraction

We investigated a variety of approaches in the feature extraction phase, which is an important aspect of strengthening the discriminative power of our models. These approaches included n-grams, Inverse Frequency-Inverse Term Frequency (IF-IFTF), and the use of word embedding models that have already been trained, such as GloVe and FastText, for Arabic news classification. Deeper comprehension of textual relationships is fostered by N-grams, which are skilled at capturing sequential word patterns and offering a nuanced representation of linguistic context. In addition, the weighted term frequency metric IF-IFTF guarantees that informative and category-specific terms are prioritized.

Moreover, our feature set is enhanced by the incorporation of pre-trained word embedding models. The global vector representation model GloVe and the sub-word information capture tool FastText provide pre-computed embeddings of Arabic language that capture syntactic and semantic relationships. Our method benefits from the transfer of knowledge from large language corpora by using these pre-trained models, which improves the models' comprehension of contextual meanings and their ability to capture semantic nuances found in Arabic news articles. This comprehensive feature set improves our models' performance in the challenging task of Arabic news categorization by strengthening their robustness and efficacy.

This paper presents a hybrid machine learning model that combines embeddings to capture contextual subtleties and term importance, and TF-IDF weighted features to improve natural language processing. These features are combined with the output of a sequential network that can comprehend sequences to further refine the model. The combined data is then sent to a Dense layer for comprehensive analysis. This method offers improvements in NLP techniques by fusing the benefits of sequential and statistical data processing.

3.4 Base classifiers

To optimize the overall predictive performance, we employ a variety of machine learning classifiers, each of which is designed to play to its unique strengths. Below is a brief outline of every classifier:

Decision Trees (DTs): DTs are tree-like models that use nodes to navigate through successive nodes until a final prediction is made. DTs are especially good at capturing nonlinear relationships and providing interpretability.

Logistic Regression (LR): Binary classification tasks frequently employ LR as it is a popular model for modeling the probability of a binary outcome using a logistic function. It is effective, easy to understand, and simple—especially when applied to problems that can be divided into linear components.

Naive Bayes (NB): is a probabilistic classifier that assumes feature independence and is based on the Bayes theorem. Its effectiveness—particularly with high-dimensional datasets—and its excellent sparse data performance are its main advantages.

Random Forest (RF): is an ensemble of decision trees that collaboratively combine their predictions to enhance accuracy and robustness. Its strengths include robustness to overfitting, adept handling of nonlinearities, and the provision of feature importance rankings.

Support Vector Machine (SVM): With the intention of maximizing the margin between classes, SVM create hyperplanes in high-dimensional spaces to divide them. Reliability with various kernel functions, robustness against overfitting, and efficacy in high-dimensional spaces are among its advantages (Schölkopf, 1998).

Linear Support Vector Machine (LSVM): A version of SVM called the Linear Support Vector Machine (LSVM) is intended for data that can be separated linearly and focuses on finding the best hyperplane for linear classification. Adaptability to large-scale datasets and efficiency in solving linearly separable problems are among its strong points.

Journal of Theoretical and Applied Information Technology
15th March 2024. Vol.102. No 5

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3.5 Deep learning models

After integrating traditional methods for feature extraction, we transitioned to more advanced deep learning architectures, employing Convolutional Neural Network (CNN) and Gated Recurrent Unit (GRU) models. Convolutional layers enable the CNN, which is well-known for its abilities in image analysis, to capture hierarchical features in sequence-based tasks with equal efficacy. Conversely, the recurrent neural network variant known as the GRU does a great job of modeling sequential dependencies while concentrating on addressing vanishing gradient problems. Our goal is to leverage the complementary strengths of these deep learning architectures for improved Arabic news categorization performance by smoothly incorporating them into our framework.

Convolutional neural network (CNN): This is a haphazard multilayer perceptron version. An input layer, an output layer, and multiple hidden layers are the standard components of a CNN. The hidden layers of a CNN typically consist of convolutional, pooling, and fully connected layers.

Long Short-Term Memory (LSTM): Specialized recurrent neural networks (RNNs) called LSTM networks are excellent at remembering information over long periods of time, which makes it possible to analyze temporal patterns in data. In order to control information flow and overcome issues such as the exploding and vanishing gradient problem that arise during deep neural network training, LSTMs make use of memory, input, output, and ignore gates.

Another kind of recurrent neural network that is comparable to an LSTM but has fewer parameters is the gated recurrent unit (GRU). Sequence classification tasks have demonstrated the effectiveness of GRU, particularly when long-term dependencies are not as significant. In our research, we classified the Arabic text data using GRU according to the features produced by various feature extraction methods.

Hybrid Networks (CNN-LSTM, and CNN+GRU): In hybrid networks, the output for the subsequent layer (LSTM, or GRU) is produced solely by applying the convolutional layer as a feature extraction layer with max-pooling.

Sequential Networks in Machine Learning: Neural networks specifically designed for processing sequential data include sequential networks. In situations like natural language processing, where comprehending the overall structure and meaning depends on the order and context of individual elements, these models are especially skilled at handling data. They are intended to more effectively capture long-range dependencies in data and are improvements over the fundamental RNN (Recurrent Neural Network) paradigm. An overview of each is provided below:

- **BiLSTM (Bidirectional Long Short-Term Memory):** An RNN type called an LSTM unit is made to retain long-term dependencies in sequential data. It accomplishes this by controlling the information flow through an intricate network of input, output, and forget gates. When using a BiLSTM, the input sequence is run through two independent LSTMs, one in the forward direction and one in the backward direction. This makes it possible for the model to include data from both the past and the future.
- **BiGRU (Bidirectional Gated Recurrent Unit):** A GRU is another kind of RNN that attempts to resolve the vanishing gradient issue of conventional RNNs, much like an LSTM. Compared to LSTMs, GRUs use two gates—the update and reset gates—instead of three, which makes them somewhat simpler. Similar to BiLSTM, BiGRU applies two distinct GRUs—one for each direction—to the input data. As a result, the network can now gather data from the past and the future.
- **XGBoost Classifiers (XGB):** The advanced ensemble algorithm known as XGBoost, or eXtreme Gradient Boosting, is highly effective and efficient in predictive modeling. It constructs decision trees in a sequential fashion, with each one fixing the mistakes of the preceding one to increase accuracy and resilience. With features like handling missing data and regularization to prevent overfitting, XGBoost excels at handling complex datasets. Because of its scalability, it is perfect for large-scale data analysis, which improves our models' predictive performance.
- **Our Hybrid model:** The three main elements of our Hybrid model's methodology are designed to improve text classification: In order to determine the meaning of each word in the corpus, it first calculates the weighted average features for each document using the TF-IDF method. Second, it uses FastText word embeddings concatenated with these TF-IDF features, a tactic that takes advantage of the syntactic and semantic relationships present in the words. Ultimately, the model trains a complex classifier that can recognize and

Journal of Theoretical and Applied Information Technology
15th March 2024. Vol.102. No 5

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

comprehend subtle patterns in the text using this expanded feature set.

- The methodology for the hybrid model encompasses two primary components: feature extraction and neural network architecture.

Let $X = \{x_1, x_2, \ldots\ldots, x_n\}$be a set of text documents and $Y = \{y_1, y_2, \ldots\ldots, y_n\}$ their corresponding labels.
Our objective is to train a model that predicts the label y given a document x.

1. Feature extraction:

Each label y_i is encoded into a unique integer value.
$y_{encoded} = f_{LE}(y_i)$

Each document x_i is tokenized and padded to a fixed length.
$x_{seq} = f_T(x_i)$
$x_{padded} = f_P(x_{seq}, maxLen)$
where the padding function, f_P, and the tokenization function, f_T, guarantee that each sequence has a fixed length, $maxLen$.

TF-IDF Vectorization: Word importance is captured by the TF-IDF model in relation to word frequency in individual documents and word rarity throughout the dataset. This method aids in highlighting words that stand out in individual articles but are not frequently used in articles as a whole, offering a nuanced feature representation helpful for classification tasks.
Each document x_i is also represented as a TF-IDF vector.
$V_i = f_{TFIDF}(x_i)$
where $V_i \in \mathcal{R}^d$ the TF-IDF vector of document x_i, and d is the number of features.

FastText Embeddings: We employed a pre-trained word embedding model called FastText, which has the benefit of incorporating sub-word information, which makes it useful for languages with complex morphology, such as Arabic.
Each word w in the vocabulary is represented by a FastText embedding.
$e_w = f_{FT}(w)$, where $e_w \in \mathcal{R}^k$ is the embedding vector of word w and k is the embedding dimension.

2. Neural Network Architecture: Sequential Networks

The model's bidirectionality enables it to process sequences in both directions, capturing context in both cases. GRUs are a kind of recurrent neural network (RNN) that work well with sequence data, like text, because they keep track of information about previous elements in the sequence inside an internal state.

- **Input Layers:** Different Representations with Different Inputs:
- **Embedded Sequences Input:** FastText-generated word embeddings, for example, are handled by this input layer. Words are transformed into dense vector representations by these embeddings, which capture syntactic and semantic information.
- **TF-IDF Vectors Input:** A statistical metric called TF-IDF is used in text classification to assess a word's significance to a document within a set of documents. This input layer receives vectors, each of whose dimensions represents a word's TF-IDF score—a measure of the word's importance within the document.
  Here, a neural network's use of distinct input layers for TF-IDF vectors and embedded sequences enables a more in-depth and thorough analysis of text data. This technique successfully combines the statistical significance indicated by TF-IDF vectors with the contextual insights offered by word embeddings, potentially producing more potent and precise text processing models.
  Sequential data-handling layers such as RNNs, LSTMs, or GRUs are frequently used to process the embedded sequences input. Understanding the linguistic structure of the text depends on these layers' ability to capture the context and order of sequences. Since the TF-IDF input is a more conventional vector representation, dense layers—fully connected layers—are usually used to process it. Compared to the contextual data from embeddings, these layers offer a different perspective by capturing the significance and frequency of words in the document.
- **Embedding Layer:**

We use an embedding matrix initialized with FastText vectors to transform the input token sequence into dense vectors. An Embedding layer, the base layer of our model, functions as a lookup table by allocating a high-dimensional vector to every word in our corpus. These vectors, which represented the semantic characteristics of words and their significance within the corpus, were pre-initialized with weights obtained from TF-IDF-scaled FastText embeddings. We made this layer non-trainable in order to protect the integrity of these pre-trained embeddings and guarantee that the acquired word relationships held true during training.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

- **Sequential Networks: BiGRU and BiLSTM Layers:**

Sophisticated in capturing contextual dependencies, the BiLSTM and BiGRU layers are essential for processing sequential data in sequential networks. The sequence of embedded words is passed through a sequential layer.
$h = SequentialLayer\left(\mathbf{E}(x_{padded})\right)$, where **E** is the embedding matrix, and h is the output of the sequential layer.

- **Concatenation Layer:**

The TF-IDF features are concatenated with Sequential Networks like BiGRU layer's outputs. In this step, the statistical word importance features from the TF-IDF vectorization are combined with the deep learning-based sequential understanding from the BiGRU layer. The model's later layers then receive this concatenated feature set. Including the word importance metrics from the TF-IDF vectors, this combined representation now includes the rich syntactic and semantic information from the word embeddings. We represent a comprehensive approach to text classification, by utilizing TF-IDF's statistical word importance analysis in combination with a neural network's sequential pattern recognition capabilities. With its strong feature representation for efficient classification, this hybrid model excels at managing the complexities of languages like Arabic. This methodology combines the best aspects of traditional NLP techniques (TF-IDF, which offers a statistical representation of word importance) with the capabilities of deep learning (through the BiGRU model, which is good at capturing sequential patterns in text data). Text classification tasks, especially those involving complex languages like Arabic, can benefit greatly from the use of this hybrid approach. In order to capture a wide range of linguistic characteristics and possibly increase classification accuracy, both sequence data and TF-IDF features are used.
The output of the sequential layer is concatenated with the TF-IDF vector.
$Concant = [h; V_i]$
For classification, a dense layer with a $sofmax$ activation is passed through the concatenated vector.
$O = sofmax(W, Concant + b)$, Where O is the output probabilities, W is the weight matrix of the dense layer, b is the bias, and $sofmax$ is the activation function.

3.6 Ensemble learning

As part of an ensemble learning approach called stacked generalization, several models are trained and their predictions are combined using a meta-model. In order to improve predictive performance, the paper investigates the concept of combining multiple models in a way that lets them learn when to defer to one another (Wolpert, 1992). Within the field of ensemble learning, combining predictions from various models is a common tactic, and voting is a key method for this blend. There are two main ways that the idea of voting can appear: hard voting and soft voting. Soft voting takes into account the weighted average of predicted probabilities, whereas hard voting combines predictions through a majority vote (Dietterich, 2000). We plan to address both stacking and voting ideas in our research paper, explaining how these methods, in addition to other ensemble approaches, help to improve prediction accuracy. In this study, we leverage the strength of ensemble learning by utilizing voting and stacking techniques to improve the predictive power and robustness of our Arabic news classification model. The group consists of different classifiers, each bringing special advantages to the group's decision-making process. The ensemble comprises various classifiers, each contributing unique strengths to the collective decision-making process. Our ensemble includes Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) layers for effective feature extraction from text data. Additionally, traditional machine learning models such as XGBoost, Random Forest, and Support Vector Machine (SVM) are incorporated, each tailored to capture distinct patterns within the dataset. Through the use of stacking and voting strategies, our research attempts to leverage complementary strengths amongst different models, resulting in a more accurate and robust Arabic news categorization model. When ensemble methodologies are combined with sensible learning, they demonstrate how different models can work in concert to improve overall predictive capabilities.

We proposed an ensemble method for text classification that combines sequential neural network models with a combination of machine learning algorithms. The classifiers in our ensemble, RandomForest, SVM, and XGBoost, are best known for their reliable performance in a range of environments. In addition, we include CNNs for their effectiveness in extracting hierarchical features and sequential networks like BiLSTM and BiGRU, which are skilled at capturing the temporal

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

dynamics of language. This combination makes use of deep learning models' understanding of textual data's sequence and context, as well as traditional algorithms' discriminative feature processing capabilities.

4. RESULTS AND ANALYSIS

In the pursuit of enhancing Arabic text classification methodologies, this study set out to achieve several key objectives. In line with the outlined research objectives, our study aimed to provide a comprehensive examination of the Arabic text classification workflow, covering preprocessing, feature extraction, and classification techniques. The impact of base classifiers is explored in detail in Section 4.1, which provides comprehensive findings for various datasets and feature extraction techniques. Section 4.2 delves deeper into the effects of hybrid models that combine different neural network architectures and word embedding techniques. This study reveals insights into the combined effectiveness of CNN, GRU, BiGRU, and Glovo embeddings with fasttext, cbow, and BiLSTM architectures for processing Arabic text. Additionally, as a novel feature to improve natural language processing, the hybrid approach—which combines embeddings with TF-IDF weighted features is presented. In Section 4.3, we extend our exploration into ensemble learning, introducing a novel and sophisticated hybrid feature stacking model.

4.1 The impact of base classifiers

The detailed results of our experiments are presented in Table 1, which summarizes the classification performance across three datasets (Akhbarona, Arabiya, Khaleej) using two feature extraction methods (TFIDF and N Grams) and six different classifiers (DT, LR, NB, RF, SVM, LSVM). The results indicate that Logistic Regression (LR), Support Vector Machine (SVM), and Linear Support Vector Machine (LSVM) consistently achieved high accuracy across all datasets and feature extraction methods, with scores ranging from 0.92 to 0.97. Random Forest (RF) also demonstrated robust performance, consistently scoring 0.90 or higher. Decision Tree (DT) and Naive Bayes (NB) showed competitive accuracy, while their performance varied across datasets and feature extraction methods. Overall, the results suggest that LR, SVM, and LSVM are promising classifiers for Arabic news categorization across diverse datasets and feature representations.

Table 1: Classification Performance Across Datasets and Feature Extraction Methods

Dataset / Classifier	Akhbarona		Arabiya		Khaleej	
	TFIDF	N Grams	TFIDF	N Grams	TFIDF	N Grams
DT	0.82	0.82	0.90	0.90	0.87	0.87
LR	0.92	0.90	0.97	0.96	0.96	0.95
NB	0.88	0.84	0.93	0.90	0.94	0.90
RF	0.90	0.90	0.96	0.95	0.96	0.95
SVM	0.92	0.91	0.97	0.96	0.97	0.96
LSVM	0.92	0.91	0.97	0.96	0.96	0.95

The impact of stemming on the classification performance across three datasets (Akhbarona, Arabiya, Khaleej) and two feature extraction methods (TFIDF and N Grams) using six different classifiers (DT, LR, NB, RF, SVM, LSVM) is summarized in Table 2. The results reveal varying effects of stemming on different classifiers and datasets. Logistic Regression (LR), Support Vector Machine (SVM), and Linear Support Vector Machine (LSVM) consistently demonstrated robust accuracy across all datasets and feature extraction methods, with scores ranging from 0.93 to 0.98. Random Forest (RF) also exhibited strong performance, consistently scoring 0.91 or higher. Decision Tree (DT) and Naive Bayes (NB) showed more varied results, with the impact of stemming varying across datasets. Table 2 provides a comprehensive overview of the classification performance under the influence of stemming, showcasing the nuances and variations in model behavior across different classifiers and datasets.

Table 2: Impact of Stemming on Classification Performance

	Akhbarona		Arabiya		Khaleej	
	TFIDF	N Grams	TFIDF	N Grams	TFIDF	N Grams
DT	0.81	0.81	0.89	0.89	0.85	0.86
LR	0.93	0.92	0.98	0.96	0.97	0.96
NB	0.88	0.80	0.93	0.87	0.92	0.89
RF	0.91	0.91	0.96	0.96	0.96	0.96
SVM	0.94	0.92	0.98	0.96	0.98	0.96
LSVM	0.93	0.92	0.98	0.96	0.97	0.95

Combining the insights from Tables 1 and 2, which detail the classification performance across various datasets, feature extraction methods, and

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

classifiers, reveals valuable patterns in Arabic news categorization. The first table underscores the consistency and high accuracy of Logistic Regression (LR), Support Vector Machine (SVM), and Linear Support Vector Machine (LSVM) across diverse datasets. Random Forest (RF) also proves to be a robust performer. In the second table, the impact of stemming on classification performance is explored, showcasing nuanced variations in model behavior. LR, SVM, and LSVM maintain their robust accuracy, with Random Forest consistently strong. Decision Tree (DT) and Naive Bayes (NB) exhibit varied responses to stemming across datasets. This combined analysis provides a comprehensive understanding of the effectiveness of different classifiers and feature extraction methods, offering valuable insights for Arabic news categorization tasks.

4.2 The impact of Hybrid Models with Different Word Embeddings

Our study focused on different word embedding techniques (fasttext, cbow, Glovo) in conjunction with neural network architectures (CNN, GRU, BiGRU, BiLSTM) and hybrid approaches to investigate the efficacy of various NLP ensemble models in processing Arabic text. Three Arabic news sources—Akhbarona, Arabiya, and Khaleej—were used to test the models in both stemming and non-stemming scenarios. Futhermore, A major factor was the word embedding selection (fasttext, cbow, Glovo), with fasttext often outperforming the other options in situations where there was no stemming. This suggests that in Arabic text analysis, embedding choice is crucial. In the majority of configurations and sources, models tended to perform better in the no stemming condition. This pattern draws attention to the possible loss of contextual data when stemming Arabic text. Although all three models executed flawlessly, Arabiya had the highest accuracy rates, indicating source-specific traits that could affect model performance. All sources demonstrated consistently better performance than the BiGRU+Hybrid, LSTM+Hybrid, and BiLSTM+Hybrid models, especially in the no stemming condition as it shown in Table 3. This implies that hybrid approaches are effective for handling the Arabic language's complexities.

It is clear that under no stemming conditions, in particular, the hybrid models—especially those that include BiGRU—produce the highest accuracy rates of any source as it illustrated in figure 1. This shows that hybrid models can handle the Arabic language's complexity better. The significance of embedding choice in model efficacy is further supported by the strong performance of models that combine CNN and GRU or BiLSTM layers with fasttext embeddings. Because of the possible loss of semantic information necessary for model interpretation and classification tasks, stemming may have drawbacks in Arabic natural language processing (NLP). This is demonstrated by the consistent superiority of the no stemming approach across nearly all models.

Table 3: Performance Evaluation of NLP Ensemble Models on Arabic News Sources.

	Akhbarona		Arabiya		Khaleej	
	Stemming / No Stemming		Stemming / No Stemming		Stemming / No Stemming	
fasttext +CNN+ GRU	0.90	0.93	0.96	0.97	0.94	0.96
cbow+CNN + GRU	0.90	0.89	0.96	0.95	0.93	0.95
Glovo+CNN+ GRU	0.91	0.89	0.96	0.96	0.94	0.93
fasttext +BiGRU	0.91	0.93	0.97	0.97	0.95	0.97
fasttext+BiLSTM	0.91	0.94	0.97	0.97	0.95	0.96
cbow+BiGRU	0.92	0.91	0.97	0.96	0.95	0.95
cbow+ BiLSTM	0.92	0.92	0.97	0.96	0.95	0.95
Glovo +BiGRU	0.92	0.92	0.97	0.97	0.95	0.94
Glovo + BiLSTM	0.92	0.91	0.97	0.96	0.95	0.94
BiGRU+Hybird	0.92	0.94	0.97	0.98	0.96	0.97
LSTM +Hybird	0.91	0.94	0.97	0.98	0.96	0.97
BiLSTM+Hybird	0.91	0.94	0.97	0.98	0.95	0.97

This table shows the accuracy of several NLP ensemble models across three Arabic news sources (Akhbarona, Arabiya, Khaleej) under stemming and without stemming conditions. The models combine different word embeddings (fasttext, cbow, Glovo) with neural network architectures (CNN, GRU, BiGRU, BiLSTM), and hybrid approaches. The outcomes show how well hybrid models work as well as how word embedding and preprocessing decisions affect model performance.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

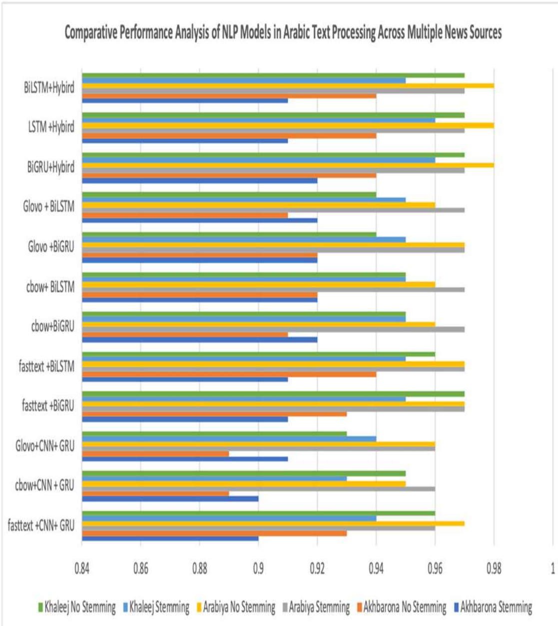

Figure 1: Comparative Analysis of NLP Ensemble Model Performances in Arabic Text Processing.
This chart compares three Arabic news sources (Akhbarona, Arabiya, and Khaleej) to show the accuracy of different NLP ensemble models using different word embedding combinations (fasttext, cbow, Glovo) and neural network architectures (CNN, GRU, BiGRU, BiLSTM), including hybrid approaches. The stemming and non-stemming conditions of the results show the effectiveness of each model in various preprocessing settings.

The confusion matrix shows that the BiGRU+Hybrid Feature Model performs impressively in terms of classification when applied to the Arabiya source as demonstrated in Figure 2. The model demonstrates remarkable precision in accurately classifying major classes, displaying remarkably elevated true positive rates, with 4623 for the first class and 5891 for the fourth class, respectively. In less represented classes, such as the second and fifth classes, where there are 833 and 774 true positives, respectively, it maintains high accuracy while handling class imbalance effectively. The low rate of misclassifications compared to the accurate predictions suggests that the model is good at differentiating between classes. The model's overall robustness and reliability in classifying a wide range of classes are highlighted by the results, highlighting its efficacy for text classification tasks in the Arabiya context.

Figure 2: The BILSTM +Hybrid Feature Model's Confusion Matrix for Arabiya Source demonstrates the model's high classification accuracy across a variety of classes.

The Confusion matrix for the BiGRU+Hybrid feature model (figure 3) shows high accuracy and robust performance in classifying various classes. The high percentage of true positives (4629 in the first class and 5868 in the fourth class) and low rate of misclassifications demonstrate the model's high degree of accuracy in a number of classes. Even though it happens infrequently, misclassification proves that there is real class discrimination. Additionally, the model's ability to handle class imbalance is evident, as it consistently shows excellent accuracy in classes with smaller sample sizes.

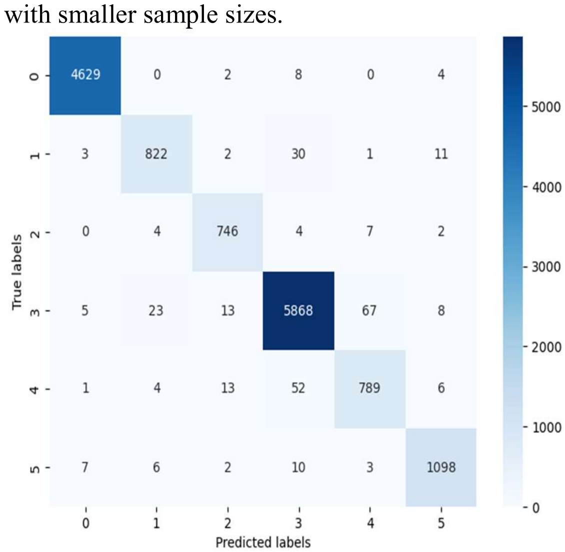

Figure 3: The BiGRU+Hybrid Feature Model's Confusion Matrix for Arabiya Source demonstrates the model's high classification accuracy across a variety of classes.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4.3 The Impact of Ensemble Learning

We evaluated four different ensemble models' performance in Arabic text analysis using Akhbarona, Arabiya, and Khaleej as news sources as it shown in table 4. In processing Arabic text data, we compare the performance of different ensemble models. Based on four main models, each with a unique approach to text analysis, this evaluation is conducted. The models are:

1. **Stacking-CNN:** Convolutional Neural Network (CNN) with Long Short-Term Memory (LSTM) layers, Support Vector Machine (SVM), XGBoost, and Random Forest algorithms are used in the stacking-CNN ensemble model. The sequential layering technique is utilized by this model to improve the accuracy of feature extraction and classification. Demonstrated excellent performance, with a peak accuracy of 0.97 in Arabiya, especially in the no stemming condition.
2. **Voting-CNN:** The CNN-LSTM, SVM, XGBoost, and Random Forest components of this model work together to make decisions. Using the advantages of each unique model, the final result is decided by a majority vote. Despite being marginally less efficient than Hybrid-Bi and Stacking-CNN, it demonstrated strong performance, particularly in situations without stemming.
3. **Stacking-Bi:** A complex ensemble that combines SVM, XGBoost, and Random Forest with Bidirectional LSTM (BiLSTM) and Bidirectional Gated Recurrent Unit (BiGRU). This stacking method processes text data both forward and backward in an attempt to more successfully capture complex language patterns. This model's best accuracy in Arabiya under no stemming condition was 0.98, which was comparable to Stacking-CNN'sperformance.
4. **Hybrid-Bi:** A sophisticated hybrid feature stacking model that integrates XGBoost, Random Forest, and SVM after combining BiLSTM and BiGRU. For better predictive performance, this model combines and refines the features that are extracted by both BiLSTM and BiGRU.

Table 4: Comparative Performance of Ensemble Models on Arabic Text Analysis Across Different News Sources

Source / Ensemble model	**Stacking-CNN**		**Voting-CNN**		**Stacking-Bi**		**Hybrid-Bi**	
	Stemming / No Stemming		Stemming / No Stemming		Stemming / No Stemming		Stemming / No Stemming	
Akhbarona	0.91	0.94	0.90	0.93	0.91	0.95	0.92	0.94
Arabiya	0.96	0.97	0.95	0.93	0.95	0.98	0.97	0.98
Khaleej	0.94	0.97	0.93	0.95	0.95	0.98	0.96	0.97

Each of the models—Stacking-CNN, Voting-CNN, Stacking-Bi, and Hybrid-Bi—was evaluated in both stemming and non-stemming scenarios. The following is a summary of the major findings:
Using BiLSTM, BiGRU, SVM, XGBoost, and Random Forest, the Hybrid-Bi model (Hybrid Feature using Stacking) consistently outperformed other models in all news sources and in both stemming and non-stemming scenarios. In the case where there was no stemming, this model had the highest accuracy, scoring as high as 0.98 in Arabiya and Khaleej. All of the models showed a noteworthy trend: overall, the no stemming condition performed better than the stemming condition. This shows that for the accuracy of ensemble models in Arabic text processing, preserving the complete form of words might be more advantageous. Furthermore, we found in our research that various news sources showed different performance trends. The Hybrid-Bi model outperformed the other models marginally for Akhbarona, with all models functioning similarly. The models from Hybrid-Bi and Stacking-Bi performed exceptionally well in the no stemming condition, while Arabiya showed the highest accuracy. The Hybrid-Bi model attained the highest accuracy in the Khaleej source, which similarly showed promising results. In Arabic text analysis, these results demonstrate the efficacy of sophisticated hybrid and stacking techniques.

The efficacy of hybrid and stacking approaches in Arabic language processing is demonstrated by these results, especially when complete lexical features are maintained. The Hybrid-Bi model's better performance in a variety of news sources and environments demonstrates its potential as a reliable tool for Arabic text analysis.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

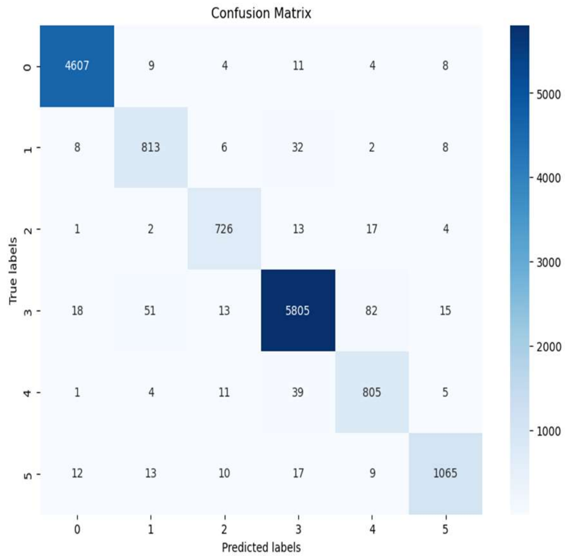

Figure 4: Hybrid-Bi Model Performance on Arabiya Dataset with stemming

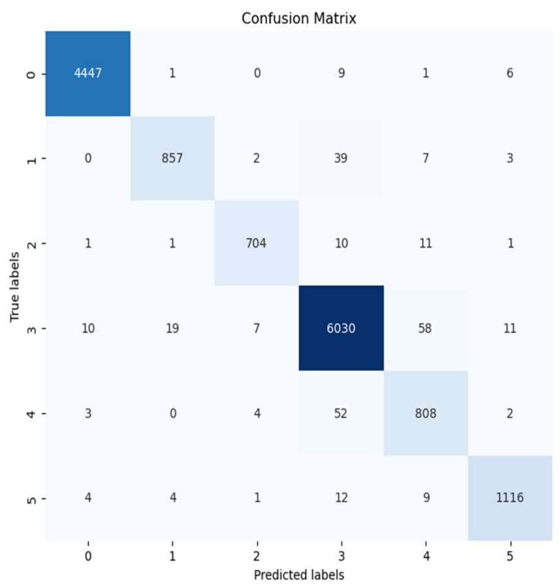

Figure 5: Hybrid-Bi Model Performance on Arabiya Dataset with no stemming

The model's high accuracy in classifying Arabic texts into six groups is reflected in the Arabiya classification report after applying stemming (figure 4). Particularly in the first category, where 4607 correct classifications were made, the confusion matrix shows a strong true positive rate. Even though a high percentage of accurate predictions are made, some categories exhibit some confusion. The fourth category, for example, has more widely distributed misclassifications. All things considered, the model proves to be a reliable means of classifying Arabic text using stemming, exhibiting very little error in its ability to distinguish between classes. On the other hand, an excellent representation of the model's accuracy can be seen in the confusion matrix for "Arabiya" without stemming as it shown in figure 5. With 4447 true positives and very few misclassifications, Category One demonstrated remarkable precision. Despite a slight increase in misclassification (39), compared to category one, the second category also performed well, with 857 true positives. With 704 accurate classifications and little confusion with other classes, the third category kept a high level of accuracy. Interestingly, category four outperformed the others in terms of true positives (6030); however, it also had a somewhat higher number of misclassifications, which might indicate a wider range of themes. With 808 and 1116 true positives, respectively, and relatively few errors, categories five and six demonstrated exceptional precision.

5. CONCLUSIONS AND FUTURE WORK

In this paper, we develop and analyze deep learning methodologies to present a major breakthrough in the field of Arabic news classification. The creation of a novel hybrid neural network that creatively combines layers of sequential networks, FastText embeddings, and TF-IDF weighting is a significant contribution of our work. Because of its unique ability to record both static and dynamic language features, this architecture improves the subtleties of natural language processing text classification. We present a thorough comparative analysis of various deep learning models, focusing on our Hybrid-Bi model in particular. Using a stacking technique, this model represents an advanced fusion of hybrid features, including SVM, XGBoost, BiGRU, BiLSTM, and Random Forest. It establishes a new standard for Arabic news classification with its consistently improved performance across multiple datasets, especially noteworthy in non-stemming scenarios. This discovery is particularly revolutionary because it contradicts the accepted knowledge in NLP about the usefulness of stemming. Peak accuracy values of up to 0.98 in the Arabiya and Khaleej sources, for example, not only establish new accuracy records but also demonstrate the efficacy of our approaches in real-world scenarios. Our results greatly advance the state of the art in Arabic natural language processing (NLP), providing new insights into the function of text preprocessing and opening up new directions for future investigation in this quickly developing field. Because of its primary focus on

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Modern Standard Arabic, the study may have overlooked the rich linguistic diversity that results from dialectal variations. In order to account for the subtleties and regional variations, future research could examine the opportunities and difficulties of integrating dialectal Arabic into NLP models.

Availability of supporting data: Data sharing is not applicable to this article as no datasets were generated during this study.
Competing interests: The author declare that they have no conflict of interest.
Authors' contributions: Author have read and agreed to the published version of the manuscript.

REFERENCES:

[1] Ababneh, A. H. (2022) 'Investigating the Relevance of Arabic Text Classification Datasets Based on Supervised Learning', *Journal of Electronic Science and Technology*. University of Electronic Science and Technology of China, 20(2), pp. 187–208. doi: 10.1016/j.jnlest.2022.100160.

[2] Ahanin, Z. *et al.* (2023) 'Hybrid Feature Extraction for Multi-Label Emotion Classification in English Text Messages', *Sustainability (Switzerland)*. Multidisciplinary Digital Publishing Institute, 15(16), p. 12539. doi: 10.3390/su151612539.

[3] Alayba, A. M. and Palade, V. (2022) 'Leveraging Arabic sentiment classification using an enhanced CNN-LSTM approach and effective Arabic text preparation', *Journal of King Saud University - Computer and Information Sciences*, 34(10), pp. 9710–9722. doi: 10.1016/j.jksuci.2021.12.004.

[4] Ameur, M. S. H., Belkebir, R. and Guessoum, A. (2020) 'Robust Arabic text categorization by combining convolutional and recurrent neural networks', *ACM Transactions on Asian and Low-Resource Language Information Processing*. ACM PUB27 New York, NY, USA, 19(5). doi: 10.1145/3390092.

[5] Bani Younes, Mohammad, Bani Younes, Mutaz and Al-khdour, N. (2022) 'Identify Arabic dialect using ensemble model'. Available at: https://doi.org/10.21203/rs.3.rs-1564007/v1.

[6] Berrimi, M. *et al.* (2023) 'Attention Mechanism Architecture for Arabic Sentiment Analysis', *ACM Transactions on Asian and Low-Resource Language Information Processing*, 22(4). doi: 10.1145/3578265.

[7] Darwish, K., Mubarak, H. and Abdelali, A. (2017) 'Arabic Diacritization: Stats, Rules, and Hacks', *WANLP 2017, co-located with EACL 2017 - 3rd Arabic Natural Language Processing Workshop, Proceedings of the Workshop*, (January), pp. 9–17. doi: 10.18653/v1/w17-1302.

[8] Dietterich, T. G. (2000) 'Ensemble methods in machine learning', *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 1857 LNCS, pp. 1–15. doi: 10.1007/3-540-45014-9_1.

[9] Einea, O., Elnagar, A. and Al Debsi, R. (2019) 'SANAD: Single-label Arabic News Articles Dataset for automatic text categorization', *Data in Brief*. Mendeley Data, 25. doi: 10.1016/j.dib.2019.104076.

[10] Elghannam, F. (2021) 'Text representation and classification based on bi-gram alphabet', *Journal of King Saud University - Computer and Information Sciences*. King Saud University, 33(2), pp. 235–242. doi: 10.1016/j.jksuci.2019.01.005.

[11] Elnagar, A., Einea, O. and Al-Debsi, R. (2019) 'Automatic text tagging of Arabic news articles using ensemble deep learning models', *ICNLSP 2019 - Proceedings of the 3rd International Conference on Natural Language and Speech Processing*, pp. 59–66.

[12] Hijazi, M., Zeki, A. and Ismail, A. (2022) 'Arabic Text Classification based on Semantic Relations', *International Journal of Mathematics and Computer Science*, 17(2), pp. 937–946.

[13] Jbene, M. *et al.* (2021) 'A Moroccan News Articles Dataset (MNAD) for Arabic Text Categorization', *2021 International Conference on Decision Aid Sciences and Application, DASA 2021*. Institute of Electrical and Electronics Engineers Inc., pp. 350–353. doi: 10.1109/DASA53625.2021.9682402.

[14] Khalifa, S., Zalmout, N. and Habash, N. (2020) 'Morphological analysis and disambiguation for Gulf Arabic: The interplay between resources and methods', *LREC 2020 - 12th International Conference on Language Resources and Evaluation, Conference Proceedings*, (May), pp. 3895–3904.

[15] Minaee, S., Azimi, E. and Abdolrashidi, A. (2019) 'Deep-Sentiment: Sentiment Analysis Using Ensemble of CNN and Bi-LSTM Models'. Available at: http://arxiv.org/abs/1904.04206.

[16] Qadi, L. Al *et al.* (2019) 'Arabic text

classification of news articles using classical supervised classifiers', in *2019 2nd International Conference on New Trends in Computing Sciences, ICTCS 2019 - Proceedings*. Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/ICTCS.2019.8923073.

[17] Rifai, H. El, Al Qadi, L. and Elnagar, A. (2021) *Arabic Multi-label Text Classification of News Articles*, *Advances in Intelligent Systems and Computing*. Springer International Publishing. doi: 10.1007/978-3-030-69717-4_41.

[18] Schölkopf, B. (1998) 'SVMs - A practical consequence of learning theory', *IEEE Intelligent Systems and Their Applications*, 13(4), pp. 18–21. doi: 10.1109/5254.708428.

[19] Wolpert, D. H. (1992) 'Stacked generalization', *Neural Networks*, 5(2), pp. 241–259. doi: 10.1016/S0893-6080(05)80023-1.

[20] Zamzami, N., Himdi, H. and Sabbeh, S. F. (2023) 'Arabic News Classification Based on the Country of Origin Using Machine Learning and Deep Learning Techniques', *Applied Sciences (Switzerland)*, 13(12). doi: 10.3390/app13127074.

[21] Zerrouki, T. (2023) 'Tashaphyne: A Python package for Arabic Light Stemming', *Journal of Open Source Software*. Open Journals, (April).