Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 12, No. 02, June 2026.

ABPC-NET: A CAPSULE-GUIDED HYBRID FRAMEWORK FOR ROBUST ARABIC-TEXT CLASSIFICATION

Baqer M. Merzah[1] and Jafar Razmara[2]

(Received: 7-Mar.-2026, Revised: 18-Apr.-2026, 10-May-2026 and 16-May-2026, Accepted: 29-May-2026)

ABSTRACT

Arabic Text Classification (ATC) remains challenging due to the Arabic language's morphological richness and semantic complexity. This paper proposes ABPC-Net, a hybrid framework integrating a frozen Arabic Transformer encoder, a Bidirectional LSTM, parallel multi-scale CNN branches and a lightweight capsule-inspired vector projection head for hierarchical feature integration. Evaluated on the SANAD dataset and its subsets (AlArabiya, AlKhaleej and Akhbarona) over five independent runs, ABPC-Net achieves mean accuracies of 97.00±0.04%, 99.14±0.10%, 98.40±0.10% and 95.59±0.12%, respectively. Under identical experimental conditions, the proposed framework consistently outperforms re-implemented frozen and fully fine-tuned AraBERT and MARBERT baselines. Cross-dataset evaluation on BBC Arabic and CNN Arabic further provides evidence of intra-domain transferability and rapid few-shot adaptability across Arabic news sources. The reported results are scoped to Modern Standard Arabic news classification.

KEYWORDS

Arabic text classification, Deep learning, Transformer models, Capsule networks, Natural-language processing (NLP).

1. INTRODUCTION

Arabic Text Classification (ATC) is a fundamental problem in the field of Natural Language Processing (NLP), as it allows a wide range of applications, from information retrieval [1,2] to sentiment analysis, fake news detection [3], among others [4]. Though deep learning (DL) has achieved significant results in this field, ATC faces a unique set of challenges due to the complex morphology of the Arabic language, as well as the scarcity of large datasets compared to the English language [5]. This has led to a significant amount of scholarly work in the quest to create a robust model that accommodates the complexities of the Arabic language.

Moreover, DL models are challenging to leverage in the Arabic language, because the natural characteristics of the language differentiate it from Indo-European languages. Arabic has a rich root and pattern morphology, where a single trilateral root such as ك- ت- ب (k-t-b) can generate many possible semantic derivatives, such as 'katib' (كاتب - writer), 'maktaba' (مكتبة - library) and 'maktub' (مكتوب - written) [6]. Given this morphological mix, both the dimensionality and sparsity of the feature space are relatively high. The language heavily uses agglutination, where prepositions, conjunctions and pronouns are merged into the word stem. A notable example provided is the token (فسيكفيكهم- fasayakfeekahum), which is broken down into syntactic units: 'fa' (then) + 'sa' (will) + 'yakfi' (suffice) + 'ka' (you) + 'hum' (them). This hybridized morphology is not compatible with normal tokenizers [7, 8]. Additionally, the widespread omission of diacritics in Modern Standard Arabic causes orthographic confusion (homographs). For instance, the unvoweled word ذهب (dhb) can mean 'gold' or 'went' depending on the context [9]. The complexities of these features make shallow models inadequate, as they require architectures that can simultaneously consider local morphological features, sequence context and hierarchical semantic structures.

Previous studies on ATC have utilized common ML approaches and simple DL frameworks, such as Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks [10]. Elnagar et al. [5] made significant contributions to the field of ATC. These contributions comprise developing the SANAD dataset of Arabic news articles to establish a comprehensive benchmark. They

1. B. M. Merzah is with Department of Computer Science, Faculty of Education, Uni. of Kufa, Iraq. Email: baqirm.merzah@uokufa.edu.iq
2. J. Razmara is with Department of Computer Science, Faculty of Mathematics, Statistics and Computer Science, University of Tabriz, Tabriz, Iran. Email: razmara@tabrizu.ac.ir

additionally proposed multiple DL architectures, among which Attention-GRU was proposed for state-of-the-art (SOTA) performance. Modern investigations build upon these baseline methods. They propose hybrid architectures; for example, Inception-CNN in combination with LSTM [4]. These models provide additional evidence about the performance of hybrid neural architectures on various textual attributes.

Existing SOTA methods heavily rely on large pre-trained Transformer models (e.g., AraBERT), which are able to learn to capture deep contextual relationships between words. Moreover, despite these advancements, an important research gap is still to be filled. Still, the prevailing model consists of fitting a basic classification head (e.g., a single dense layer) to the output of the Transformer. Although this step is computationally efficient, it may be a bottleneck, since it cannot fully leverage the rich, high-dimensional representations from the Transformer and it cannot fully characterize the complex hierarchical relationships between the extracted features. The novelty of ABPC-Net lies in addressing this gap through three deliberate and complementary design decisions. First, a BiLSTM layer re-encodes transformer sequence outputs to capture long-range sequential dependencies with directional awareness suited to Arabic morphological structure. Second, a parallel multi-scale CNN module with kernel sizes of 2, 3 and 4 performs explicit n-gram feature extraction at multiple granularities, a design specifically motivated by Arabic agglutination. Third, a Capsule-inspired Vector Projection Head replaces the conventional scalar softmax classifier with a vector-based encoding mechanism that preserves multi-dimensional feature relationships, enabling richer integration of multi-scale representations. Unlike recent transformer-based models, such as Tasneef [11], CLGNet [12] and ABTM [13], which either attach simple dense layers or augment transformers with frequency-based features, ABPC-Net explicitly encodes hierarchical spatial relationships among contextualized features. Though a Transformer may decide what words are relevant in context, this feature-based approach may still not explicitly model how the contextualized features group together to form higher-level abstract concepts, something important for nuanced classification. To cope with this limitation, we explore using more sophisticated downstream architectures able to better interpret the rich representations generated by Transformers.

This work presents a structured downstream architectural design for transformer-based ATC and provides systematic empirical evidence that such a design can extract substantially more value from a frozen Arabic transformer encoder than full fine-tuning with a shallow classification head. The contribution is therefore primarily architectural and empirical in nature, supported by controlled experimental analysis. The main contributions of this paper are:

- Structured Downstream Architecture: We propose ABPC-Net, a structured downstream pipeline combining BiLSTM sequential re-encoding, parallel multi-scale CNN branches (kernel sizes 2, 3 and 4) and a lightweight capsule-inspired vector projection head atop a frozen Arabic transformer encoder. Their systematic integration and controlled evaluation provide new empirical insights for Arabic text classification.
- Mechanistic Analysis of Capsule-inspired Projection: We provide an ablation-driven analysis showing that the capsule-inspired projection behaves as a relational feature fusion mechanism the effectiveness of which depends on input structural richness. In particular, it degrades performance without BiLSTM, but consistently improves performance when preceded by sequential encoding, offering practical design insights.
- Comprehensive Empirical Validation: Experimental evaluation of SANAD is conducted extensively, along with source-specific performance analysis (Akhbarona, AlArabiya, AlKhaleej).
- Cross-dataset Generalization: We evaluate ABPC-Net on BBC Arabic and CNN Arabic through zero-shot transfer and few-shot domain adaptation protocols, characterizing the model's intradomain transferability across Arabic news sources.

Our results demonstrate that our model performs well in multiple news contexts and that we include a qualitative error analysis to show how the model behaves across different news domains. Our results show that the proposed hybrid architectures provide strong empirical performance, illustrated by the Capsule Networks' performance of the model, which means that it performs significantly better than the baseline. This work contributes a structured hybrid framework for Arabic news classification that demonstrates strong empirical performance on the SANAD benchmark and offers a methodological foundation for exploring multi-component architectures in Arabic NLP tasks. The remainder of this

paper is organized as follows. Related work is described in Section 2. The ABPC-Net architecture, along with the experimental settings and hyper-parameters, is described in Section 3. The results and discussion, including the ablation analysis, error analysis and limitations, are presented in Section 4. Finally, Section 5 concludes the paper and outlines future directions.

2. RELATED WORK

Arabic-text classification (ATC) has attracted tremendous emphasis in recent years and various tools were explored, such as classical machine learning (ML), especially classical approaches and advanced DL architectures. In this section, we present a review of how these techniques evolved. Among advanced ATC approaches, standard classical ML methods were widely applied, often used together with several feature-engineering techniques, such as TF-IDF and Bag-of-Words [14], including shallow-learning approaches for tagging Arabic news articles [15]. The field was transformed by DL. In the earliest DL works, it generally limited the process to processing base architectures (CNNs) to extract local features and recurrent neural networks (RNNs) (Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRUs)) for capturing sequential dependencies in text [16]. Alsaleh and Larabi-Marie-Sainte presented a hybrid model [17], on a CNN and Genetic Algorithm (GA) of ATC, representing the trend. However, their GA-CNN model, based on GloVe as the word embedding tool obtained relatively high accuracy on Moroccan and Saudi Newspaper Article Datasets, though it stated that the training time was prolonged.

In recognition of the fact that, while CNNs and RNNs fail to adequately capture the complexities involved in Arabic-text processing, subsequent research has focused on the development of more advanced hybrid architectures. This research direction, of which the present work is a part, focuses on the synergistic integration of the capabilities offered by different architectures. For example, in research that focused on the exploration of supervised-learning strategies, Ameur et al. [18] integrated the capabilities offered by different word embeddings, including static, dynamic and fine-tuned word embeddings, into RNNs and CNNs. The results obtained by the model, especially the integration of CNNs with Bidirectional Gated Recurrent Units (BiGRUs), indicated a significant level of effectiveness, as the F-score increased by as much as 98.61%. In similar research, Jamaleddyn et al. [12] proposed a novel multi-channel deep-learning model known as CLGNet, which integrates the capabilities offered by CNN, long short-term memory networks and Gated Recurrent Units. The results obtained by the model, following the extensive pre-processing and SMOTE-based balancing of the CNN, BBC and OSAC datasets, indicated a significant level of performance, as the model performed better than the capabilities offered by different deep-learning architectures.

The large-scale publicly available dataset SANAD was introduced by Elnagar et al. [5], which was an important milestone in the field. Their pioneering work has given an invaluable resource to the task and set a very high benchmark for many types of DL models, such as various hybrid CNN-RNN architectures and attention-based paradigms. They found that models incorporating attention, such as Attention-GRU, can achieve SOTA performance, thus raising the benchmark of the task. In recent times, Alnagi et al. [4] proposed a hybrid model by incorporating the Inception-CNN and LSTM layers. This further emphasizes the adoption of various neural architectures for the processing of textual features on multiple scales. Accuracies of 92% and 96% were reported for the SANAD and AlKhaleej datasets, respectively, by utilizing complex variants of the CNN algorithm.

Jalil and Aliwy [19] advanced a novel hybrid CNN-BiLSTM architecture to facilitate taskful workloads, like topic classification, sentiment analysis, emotion recognition and sarcasm detection. Combining convolutional layers for local feature extraction with LSTM units that bidirectionally describe and capture the contextual dependencies, the model performs well on embedding spatial and sequential representations. In general, their experimental results showed good performance on topic classification (97.58%) and sarcasm handling (97%), sentiment analysis (86%) and emotion recognition (81.6%). This observation demonstrates the ability of hybrid DL architectures to model much greater language complexity and variability within the context of Arabic social-media content that faces multiple challenges, from the limited length of text to informal-language use to implicit and contextual semantic cues in the field. Novel hybrid-based paradigms have further developed the state-of-the-art by exploiting rich context embeddings to their limit.

Hossain et al. [13] proposed a hybrid model known as Attention-based Transformer Model (ABTM),

consisting mainly of deep contextual data with traditional statistical features (e.g., TF-IDF and Bag-of-Words). The present study has delivered significant improvements in performance, realizing full and best-in-class performance of 97.69% accuracy on their Arabic news dataset. This is a clear indication of the growing realization of the effectiveness of combining context features with other architectural components. Along with these architectural advancements, other research has also focused on the importance of feature representation and hyper-parameter optimization. To cite an example, B. Al-onazi et al. [20] presented a hybrid model referred to as the CRNN model, which combines CNN and RNN architectures. However, the novelty in their model is the application of the Crow Search Algorithm for hyper-parameter optimization of their model. Even though their model has clearly demonstrated the importance of hyper-parameter optimization in improving model performance, it still relies on traditional features, such as TF-IDF, which do not fully exploit the power of deep context features in natural languages.

Recent SOTA mostly converges with large pre-trained Transformer models, such as AraBERT, GigaBERT and MARBERT, to model the subtle semantic structures. Yet, as noted above, a well-known limitation is that many of the applications embed a simple, shallow classifier head on the Transformer that does not take advantage of its rich, multi-layered representations. And recent works, such as the work by Hossain et al. [13], have attempted to augment Transformers with classical statistical features (TF-IDF, BoW) and these approaches continue to employ frequency-based representations that lack structural depth. In contrast, ABPC-Net introduces a structured downstream pipeline, BiLSTM for sequential re-encoding, parallel CNNs for multi-granularity local-feature extraction and a capsule-inspired vector projection for hierarchical feature integration, that qualitatively differs from the shallow-classification strategies employed by Tasneef [11], CLGNet [12] and ABTM [13]. Unlike these approaches, ABPC-Net explicitly encodes spatial and hierarchical relationships among contextualized features, preserving part-whole relationships characteristic of Arabic morphology for more robust and granular classification. Although capsule-based models have been explored in text classification, their behavior within hybrid transformer-based pipelines for Arabic-text classification has received limited systematic analysis. The present work contributes to filling this gap by providing a controlled ablation-driven analysis of how capsule-inspired projections interact with sequential and convolutional components in the Arabic-text classification setting.

3. MATERIALS AND METHODS

In this section, the proposed ABPC-Net model for the classification of Arabic news articles will be delineated.

The methodology is based on the development of a hybrid deep-learning model, which combines the power of the pre-trained transformer model, recurrent neural network and parallel capsule network in a complementary fashion. There are four main steps in the proposed methodology: (1) Data Source, which describes the SANAD dataset; (2) Data Pre-processing, which describes the pre-processing steps for the data; (3) Model Architecture, which describes the ABPC-Net model; and (4) Training and Experimental Setup, which describes the settings used for the model's training process. A block diagram for the ABPC-Net model is presented in Figure 1.

3.1 Dataset

The SANAD dataset [5] is a large-scale Arabic news corpus used for single-label text classification. Even though the raw dataset consists of around 200,000 articles, their authors constructed a refined and balanced sub-set to minimize class imbalance and remove noisy or super brief texts. In this study, the pre-processed version was used based on their established benchmark. Consequently, the dataset used for our experiment consists of 110,900 high-quality articles distributed across the Al-Arabiya, AlKhaleej and Akhbarona portals: SANAD covers seven topical categories of Culture, Finance, Medicine, Politics, Religion, Sports and Technology. The dataset itself is balanced at the category level within each source-specific sub-set of these pieces, providing equal opportunities to evaluate classification models. In addition, SANAD is further divided over training and test partitions that are used in our demonstrations. The scale, linguistic consistency (MSA) and categorical diversity of the dataset, is significant enough to serve as a benchmark for DL-based ATC systems. The category-wise distribution of SANAD for its three source-dependent classes is shown in Table 1.

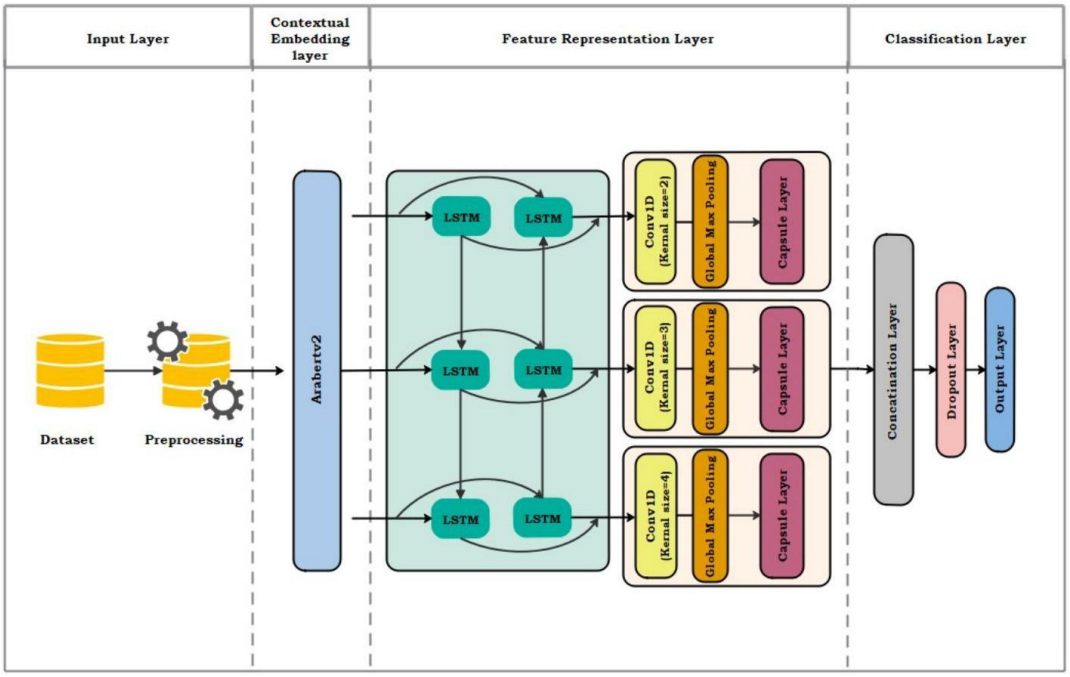

Figure 1. The ABPC-Net architecture, illustrating the sequential flow from frozen AraBERT embeddings through BiLSTM re-encoding, parallel CNN-capsule feature extraction and final classification *via* margin loss.

In particular, Al-Arabiya has five categories; Al-Khaleej and Akhbarona have all seven categories. We used the standard train-test split from the SANAD dataset, but 10% of the training dataset was allocated randomly for validation purposes. This internal validation split was used just to check how the model was performing during training and for triggering the Early Stopping mechanism. Thus, during all stages of training and hyper-parameter tuning, the official test set was solely unseen, so that at the end of testing the final evaluation was entirely unbiased.

Table 1. A balanced sub-set of SANAD articles and category counts per dataset.

Source	Categories	Training	Testing	Total	Per Category
alarabiya.net	5	16,650	1,850	18,500	3,700
alkhaleej.ae	7	40,950	4,550	45,500	6,500
akhbarona.com	7	42,210	4,690	46,900	6,700

3.2 Data Pre-processing

The text data contains noise, such as punctuation, numbers, non-Arabic scripts and diacritics, known as Tashkeel, which can negatively impact the performance of the model. Before feeding the text data into the model, a thorough and carefully tailored pre-processing plan for Arabic language is used. The major steps, which are explained in detail, are as follows:

- Normalization: Due to the fact that digital texts feature orthographic inconsistencies, a very indepth normalization is carried out. It is a crucial step, especially when the model is going to depend mainly on the semantics rather than the spelling. Among other things, this means removing all Tashkeel and Tatweel (character elongation) and unifying different forms of the Hamza (إ،أ،آ) to one form (ا). Moreover, 'Ta marbuta' (ة) is converted into 'Ha' (ه) and 'Alef maksura' (ى) is converted into 'Ya' (ي).
- Noise Removal: Repetitive and non-informative pieces are taken out of the text. This means to remove all punctuation marks, numerical digits and any Latin characters.
- Tokenization: The text after it is cleaned is broken down into single words using the Farasa

package, which is adjusted and optimized for Arabic text.

- Stop-word Removal: Concentrating on the functionality only, typical Arabic stop words, such as (e.g., من, في, على), which hardly provide any semantic value for classification, are removed using a pre-defined list from NLTK's Arabic corpus.

Such a strict and detailed pre-processing is our assurance that the data fed into our model network is cleansed, standardized and oriented towards semantically meaningful content.

3.3 Hybrid Model Architecture

Our proposed model is an end-to-end DL model designed to recognize intricate linguistic patterns. It combines the contextual capabilities of the transformer model, the sequential capabilities of the recurrent neural network and the hierarchical feature detection of the capsule network. Figure 1 illustrates the end-to-end architecture of ABPC-Net. The model processes input text through four sequential stages: (1) contextual embedding *via* frozen AraBERT, (2) sequential re-encoding *via* BiLSTM, (3) parallel multi-scale feature extraction via three independent CNN-Capsule branches with kernel sizes of 2,3 and 4 and (4) feature fusion and classification *via* concatenation and margin loss.

3.3.1 Transformer-based Embedding Layer

Our model builds upon the pre-trained Arabic transformer aubmindlab/bert-base-arabertv2 [21], which is a well-performing BERT model trained on an extensive dataset of Arabic text. Pre-processed text is tokenized with AutoTokenizer to obtain input_ids and attention_mask. The AraBERT encoder is kept fully frozen throughout all training stages (trainable = False), serving as a static contextual feature extractor. This design choice is justified on two complementary grounds. First, freezing the transformer prevents catastrophic forgetting of the broad linguistic knowledge encoded during pre-training on $\sim$ 77 GB of Arabic text, which would otherwise be at risk when fine-tuned on the comparatively smaller SANAD corpus ($\sim$ 100 K samples) [21]. Second, a frozen encoder provides a controlled experimental setting in which the contribution of the proposed downstream architecture, BiLSTM, parallel CNNs and Capsule projection, can be evaluated independently of the transformer representations, yielding a cleaner and more interpretable ablation framework. The empirical validation of this design choice is presented in Section 4.

3.3.2 Bidirectional LSTM Layer

After AraBERT generates contextualized embeddings, these sequences are fed into a recurrent neural network architecture that aims to capture longer-range dependencies as well as sequential constructs. Specifically, the full per-token output sequence of AraBERT, of shape (batch_size $\times 256 \times 768$), corresponding to the last hidden states of all 256 input tokens, is passed directly into the BiLSTM without any prior pooling, CLS token extraction or dimensionality reduction. This design preserves the complete positional and contextual information across all token positions, enabling the BiLSTM to model sequential dependencies that would otherwise be lost under aggregation. Long Short-Term Memory (LSTM) units are a kind of Recurrent Neural Network (RNN) set specifically for solving the vanishing gradient of typical RNNs. The LSTM units can learn a certain internal state at the cell level and through different gates (input, forget and output) is able to select to retain or forget content for a long amount of time. Because of their ability to store and retrieve information for arbitrary durations through gating mechanisms, LSTMs are well-suited for tasks involving sequential data, especially when temporal dynamics need to be modeled. We employ a BiLSTM in our model. This architecture enhances the standard LSTM by processing the input sequence in both forward and backward directions [22]. The outputs in both directions are combined, so that each word is more enriched than the words above it in light of both the preceding and following context.

3.3.3 Parallel Convolutional and Capsule Layers

The core innovation of our model lies in the parallel feature-extraction component, which is designed to effectively capture the different levels of textual features present in the input data. The sequence of hidden states generated by the BiLSTM layer is fed in parallel to the three parallel convolutional blocks for the feature-extraction process. Each convolutional block operates independently: the BiLSTM output is fed in parallel to three Conv1D layers (kernel sizes 2, 3 and 4), each followed by its own Global Max

Pooling layer and a dedicated Capsule projection layer. The three resulting capsule tensors - each of shape $(N \times D)$ where N is the number of target classes and $D = 16$ is the capsule dimension - are subsequently concatenated along the last axis, yielding a combined representation of shape ($N \times 3D$). This design ensures that bigram, trigram and quadrigram features contribute distinct vector representations to the final classification, rather than being merged prior to capsule projection. Each block consists of the following components:

- 1D Convolutional Layer (Conv1D): A Conv1D layer with kernel sizes of 2, 3 and 4 for the three respective parallel blocks [23] operates as an n-gram detector. It slides a filter over the sequence to identify various local patterns and features, such as adjacent word pairs (bigrams), trigrams and quadrigrams. The use of different kernel sizes allows for the extraction of features at multiple granularities.
- Global Max Pooling Layer: Following the Conv1D operation, the Global Max Pooling operation is applied to the output of the Conv1D layer. This layer aggregates the strongest feature present in the entire sequence, obtained by any filter used in the Conv1D operation, to capture the strongest local indicator for the particular feature. This layer represents a dimensionality-reduction operation, capturing the strongest n-gram features, which are then represented by the capsule layers in the form of vector representations for modeling the class-specific properties.
- Lightweight Vector-based Projection Head (Capsule-inspired): To increase representational expressiveness beyond standard scalar classification heads, we introduce a lightweight vector-based projection head inspired by capsule network principles [24]. We wish to be precise: this component is not a canonical capsule network, it involves no dynamic routing, no agreement mechanism and no part-whole relationship modeling in the sense of Sabour et al. [24]. Rather, it performs a deterministic learned linear projection of pooled CNN features into structured $N \times D$ vector representations, followed by a squashing non-linearity that bounds vector magnitudes while preserving directional information. The key distinction from a standard Dense classification head is that this projection produces multi-dimensional class-specific vectors rather than scalar logits, enabling vector-length-based class activation that preserves richer feature structure across the three parallel CNN branches.

Formally, given an input feature vector $\mathbf{x} \in \mathbb{R}^d$, the capsule layer performs a learned linear projection defined as:

$$\mathbf{u}_{\text{flat}} = \mathbf{x}\mathbf{W} \tag{1}$$

where $\mathbf{W} \in \mathbb{R}^{d \times (N \cdot D)}$ denotes the trainable weight matrix, N represents the number of capsules corresponding to the number of target classes and D defines the dimensionality of each capsule. The resulting projection $\mathbf{u}_{\text{flat}}$ is subsequently reshaped into structured capsule vectors:

$$\mathbf{U} = \text{reshape}(\mathbf{u}_{\text{flat}}, (N, D)) \tag{2}$$

To preserve vector magnitudes while maintaining directional information, a squashing nonlinearity is applied such that capsule lengths are bounded and interpretable:

$$\mathbf{V}_i = \frac{\|\mathbf{U}_i\|^2}{1 + \|\mathbf{U}_i\|^2} \frac{\mathbf{U}_i}{\sqrt{\|\mathbf{U}_i\|^2 + \epsilon}} \tag{3}$$

where $\mathbf{U}_i$ and $\mathbf{V}_i$ denote the i-th capsule before and after squashing, respectively. This non-linearity allows vector lengths to encode the strength of class activation while preserving multi-dimensional semantic information in vector orientations. Compared to conventional softmax classifiers that compress representations into scalar probability values, capsule vectors retain richer feature structures by maintaining multi-dimensional embeddings. This property enables the model to preserve semantic associations discovered during feature extraction, particularly when integrating outputs obtained from multi-scale convolutional branches. Furthermore, the deterministic projection mechanism eliminates the instability and computational overhead associated with routing iterations, resulting in more stable gradient behavior and reduced training cost. Final class predictions are derived from capsule vector lengths, $y_i = \|\mathbf{V}_i\|$, which aligns with capsule network principles and provides an interpretable vector-length-based decision mechanism. When BiLSTM precedes the CNN-Capsule pipeline, the hidden states, it produces encode directional and contextual dependencies across all token positions; even after

CNN and Global Max Pooling, the resulting vector retains structured inter-feature relationships that the capsule projection can meaningfully encode into vector-length-based class activations - a representation that is particularly suited to Arabic-text classification, where overlapping feature distributions across semantically similar categories (e.g., Politics *vs.* Finance, Tech *vs.* Finance) benefit from multi-dimensional encoding rather than scalar logit compression. In contrast, without BiLSTM, Global Max Pooling reduces the CNN output to a bag of independently selected scalar activations, discarding all sequential and relational structures. The capsule layer therefore functions as a relational feature-fusion mechanism the effectiveness of which is conditioned on the structural richness of its input.

The decision to maintain strict branch independence prior to capsule projection is grounded in three principled considerations. First, each kernel size (2, 3, 4) is designed to capture a structurally distinct linguistic granularity: bigrams can capture short morphological clitic combinations, trigrams can capture stem-affix patterns and short collocations and quadrigrams can capture longer phrasal units. Allowing cross-branch interaction prior to capsule projection would force the network to learn a single shared representation across these granularities, weakening the inductive bias that motivates the multi-scale design. Second, the capsule projection encodes directional information among feature dimensions through the learned weight matrix W; training this matrix on a structurally homogeneous within-branch input yields coherent within-granularity directional patterns, whereas mixing kernel sizes prior to projection forces the matrix to encode incompatible directional patterns simultaneously, weakening the relational signal that vector-based capsule representations are designed to preserve. Third, strict branch independence prior to fusion is the standard design pattern in multi-channel CNN architectures for text classification originating with Kim [23] and adopted in subsequent multi-scale text classifiers; our design extends this convention by inserting an independent capsule projection per branch before concatenation. The capsule-projection procedure is summarized in Algorithm 1.

Algorithm 1: Capsule-inspired Vector Projection

Input: Feature vector $x \in \mathbb{R}^d$ obtained from Global Max Pooling

Data: Learned projection matrix $\mathbf{W} \in \mathbb{R}^{\boldsymbol{d}\times(\boldsymbol{N}\times\boldsymbol{D})}$

Output: Squashed capsule matrix $\boldsymbol{V} \in \mathbb{R}^{N\times D}$

// Step 1: Linear projection to capsule space

$\mathbf{u}_{flat} = \mathrm{x}\mathbf{W}$

// Step 2: Structural reshaping into *N* class capsules

$\mathbf{U} = \mathrm{reshape}(\mathbf{u}_{flat}, (N, D))$

// Step 3: Non-linear capsule squashing

for $i \leftarrow 1\ to\ N$ **do**

$s_i = \|\mathrm{U}_{\boldsymbol{i}}\|^{\mathbf{2}}$ // Squared norm of the i[th] capsule

$\mathrm{V}_i = \frac{s_i}{1+s_i} \cdot \frac{\mathrm{U}_i}{\sqrt{s_i+\epsilon}}$ // Stable squashing function

end

return V

3.4 Classification Layer

Following the parallel feature-extraction pipeline, the three branch-specific capsule tensors, each of shape $(N \times D)$, where N is the number of target classes and $D = 16$ is the capsule dimension, are concatenated along the capsule-dimension axis to form a unified representation of shape ($N \times 3D$). To prevent overfitting, this concatenated tensor is passed through a Dropout layer with a rate of 0.3 applied along the feature axis. The actual fusion across kernel sizes is performed by a Lambda layer that

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 12, No. 02, June 2026.

computes the Euclidean L_2-norm along the capsule-dimension axis, reducing the ($N \times 3D$) tensor to an N-dimensional class-activation vector. This norm operation aggregates the squared contributions of all $3D$ dimensions per class, treating the bigram, trigram and quadrigram capsule sub-vectors as additive evidence sources that jointly determine each class's activation magnitude. The resulting class-activation vector is supplied directly to the margin-loss function (with $m^+ = 0.9, m^- = 0.1$ and $= 0.5$), which trains the network, so that the correct-class capsule norm exceeds m^+while incorrect-class norms remain below m^-. At inference time, the predicted class is determined by arg max over the N capsule norms. The Euclidean norm provides a parameter-free, magnitude-preserving fusion that respects the vector-based semantics of capsule representations and aligns directly with the margin-loss framework.

3.5 Experimental and Hyper-parameter Settings

We performed all the experiments utilizing the TensorFlow and Keras DL frameworks in a Google Colab environment powered by an NVIDIA GPU. For the text analysis, we used the pre-trained Bert-Basearabertv 2 model. To allow efficient computation and use pre-learned linguistic features, we froze the BERT layers and used them as a static feature extractor. The input sequences were tokenized and padded to a maximum length of 256 tokens. The architecture of the $ABPC$-Net model consists of a Bidirectional LSTM (BiLSTM) layer with 128 units, followed by three parallel 1D-Convolutional layers. These layers utilize kernel sizes of 2, 3 and 4, respectively, with each employing 64 filters to effectively capture multi-scale n-gram features. A series of parallel Capsule layers was configured, each with a vector dimension of 16, ensuring consistent representational capacity across all feature-extraction branches. For model optimization, the Adam optimizer was employed with a learning rate of 0.001.

To improve class separability, we used a customized margin-loss function (with $m^+ = 0.9, m^- = 0.1$, $\lambda = 0.5$). Specifically, this loss function penalizes class capsules with low magnitudes for the correct class or large magnitudes for incorrect classes, pushing the model to learn more discriminative and distinct feature boundaries. The training process was limited to a maximum of 10 epochs with a batch size of 32. To prevent overfitting, an Early Stopping mechanism was implemented with a patience of 3 epochs, monitoring validation accuracy. Detailed hyper-parameter values for the experimental evaluation are summarized in Table 2.

Table 2. Hyper-parameters used in the ABPC-Net model.

Hyper-parameter	Value
Transformer	AraBERT v2
AraBERT Encoder	Trainable = False
Max Sequence Length	256
Batch Size	32
Epochs	10
Optimizer	Adam
Learning Rate	0.001
BiLSTM Units	128
CNN Filters	64
CNN Kernel Sizes	2, 3, 4
Capsule Dimension	16
Loss Function	Margin Loss

To ensure statistical reliability, all experiments involving ABPC-Net and the fine-tuned baseline models were repeated over five independent runs with different random seeds. Results are reported as mean accuracy ± standard deviation. The five random seeds used for the independent runs are 42, 123, 456, 789 and 2024. From a computational perspective, ABPC-Net comprises approximately 136 M total parameters, of which only ~ 1.09M are trainable, while the remaining ~ 135M correspond to the frozen AraBERT encoder. This design substantially reduces the trainable parameter budget relative to full fine-tuning approaches. The frozen AraBERT encoder occupies approximately 540 MB of GPU memory,

which constrains achievable batch sizes on consumer GPUs. Inference under GPU batched conditions averages 1.97 ms per sample, which is suitable for asynchronous applications, such as news categorization, content moderation and information retrieval. Under CPU inference, however, latency rises substantially, which may limit applicability for synchronous real-time pipelines requiring sub-10 ms response. Scaling ABPC-Net to larger workloads or stricter latency budgets would benefit from model distillation, quantization or substitution of the BiLSTM with a lighter temporal convolutional network.

4. RESULTS AND DISCUSSION

The ABPC-Net model was trained and evaluated using Akhbarona, AlArabiya, AlKhaleej and SANAD datasets. Model performance was evaluated using Accuracy, Precision, Recall and F1-score [25]. Figure 2 illustrates the training and validation behavior of ABPC-Net across all datasets. A consistent gap between training and validation accuracy is observed, particularly on SANAD and Akhbarona.

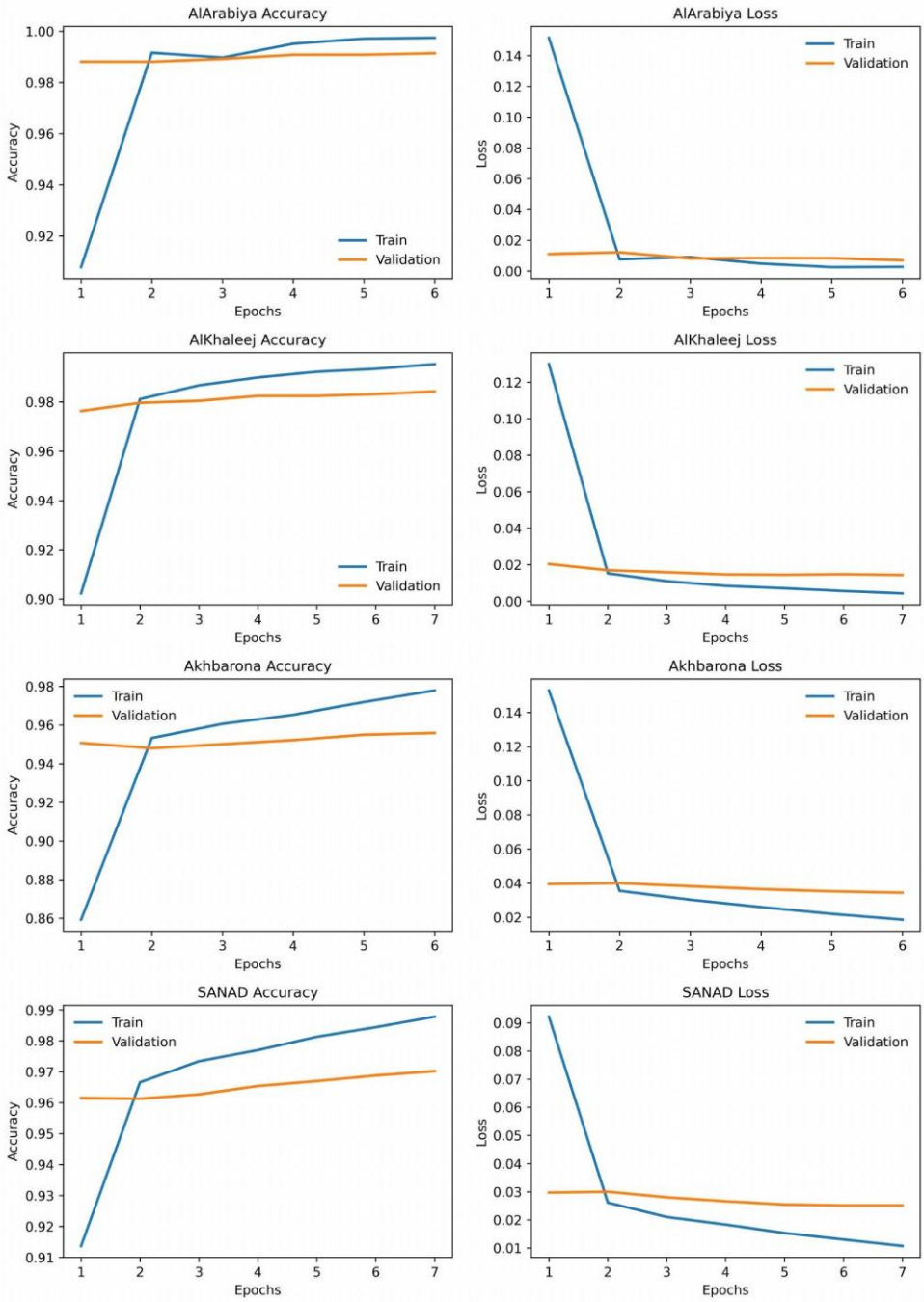

Figure 2. Training and validation performance across AlArabiya, AlKhaleej, Akhbarona and SANAD datasets. Note: Curves correspond to a single run of ABPC-Net.

This behavior is primarily attributable to two architectural factors rather than systematic overfitting: the Dropout layer (rate $= 0.3$) is active during training, but disabled during validation and the margin-loss function enforces stricter class boundary constraints during training (requiring capsule activations ≥ 0.9 for correct classes and ≤ 0.1 for incorrect classes) than are reflected in standard validation accuracy. Validation accuracy remained stable across epochs and the Early Stopping mechanism (patience $= 3$) restored the best-performing weights. The five-run statistical analysis further confirms the stability of validation performance ($\leq 0.12\%$ across all datasets). We compared ABPC-Net against four baseline configurations: frozen MARBERT, frozen AraBERT, MARBERT Full FT and AraBERT Full FT, each using a Dense classification head. All experiments involving ABPC-Net and the fine-tuned baselines were repeated over five independent runs using identical random seeds and experimental settings. Results are reported as mean ± standard deviation and paired-sample t -tests on the per-run accuracies confirmed that ABPC-Net's improvements are statistically significant ($p < 0.001$) on every evaluated dataset. Table 3 summarizes the comparative results under identical pre-processing, dataset splits and hardware conditions for directly re-implemented baselines (frozen MARBERT, frozen AraBERT, MARBERT Full FT, AraBERT Full FT).

Table 3. A comparative analysis of accuracy between ABPC-Net model and baseline models.

Methods	AlArabiya	AlKhaleej	Akhbarona	SANAD
MARBERT (frozen)	95.63 ± 0.13	95.78 ± 0.09	91.15 ± 0.11	90.60 ± 0.05
AraBERT (frozen)	96.79 ± 0.09	95.94 ± 0.10	91.74 ± 0.12	94.20 ± 0.06
MARBERT Full FT	95.98 ± 0.24	94.60 ± 0.10	90.23 ± 0.11	92.19 ± 0.04
AraBERT Full FT	97.52 ± 0.06	96.72 ± 0.10	93.49 ± 0.13	94.71 ± 0.05
ABPC-Net (ours)	$\mathbf{99.14 \pm 0.10}$	$\mathbf{98.40 \pm 0.10}$	$\mathbf{95.59} \pm 0.12$	$\mathbf{97.00 \pm 0.04}$

*All values are reported in percentages (%).
**The bold values indicate the high-accuracy performance achieved in each comparison.

On the aggregated SANAD dataset, ABPC-Net achieves $97.00 \pm 0.04\%$, surpassing the strongest fine-tuned baseline, AraBERT Full FT ($94.71 \pm 0.05\%$), by a margin of 2.29% and outperforming frozen AraBERT ($94.20 \pm 0.06\%$) by 2.80%. The consistently low standard deviation of ABPC-Net ($\leq$ 0.04% on SANAD) underscores the reproducibility of the proposed architecture. A similar pattern is observed across all individual sub-sets: ABPC-Net achieves $99.14 \pm 0.10\%$ on AlArabiya, $98.40 \pm 0.10\%$ on AlKhaleej and $95.59 \pm 0.12\%$ on Akhbarona, outperforming all baseline configurations in each case.

A particularly noteworthy finding is that ABPC-Net with a frozen AraBERT encoder consistently outperforms AraBERT Full Fine-tuned, which updates all 135 M transformer parameters with a simple Dense classification head, by margins of $2.29\%, 1.62\%, 2.10\%$ and 1.68% on SANAD, AlArabiya, Akhbarona and AlKhaleej, respectively. This result indicates that the performance gains of ABPC-Net are attributable to its structured downstream architecture, comprising BiLSTM sequential re-encoding, parallel multi-scale CNNs and a Capsule-inspired vector projection head, rather than to transformer fine-tuning. This empirical evidence further supports the frozen encoder design choice introduced in Sub-section 3.3.1, indicating that the structured downstream architecture compensates for what full fine-tuning achieves with a simple classification head. Furthermore, MARBERT Full FT ($92.19 \pm 0.04\%$ on SANAD) underperforms even frozen AraBERT ($94.20 \pm 0.06\%$), confirming that full fine-tuning of a larger transformer does not guarantee superior performance when the classification head lacks structural depth.

The class-wise performance metrics are detailed in Table 4, corresponding to a representative run selected based on its proximity to the mean accuracy across five independent runs. High precision and recall values across all categories confirm that the model maintains strong per-class discriminability, with particularly robust performance on Sports and Religion categories ($\geq$ 98% F1-score across all datasets) and greater variability in semantically overlapping categories, such as Politics and Finance, as further analyzed in Sub-section 4.4.

Table 4. Classification metrics for SANAD, AlArabiya, AlKhaleej and Akhbarona datasets.

Dataset	Class	Precision	Recall	F1-score
AlArabiya	Finance	99.18	98.11	98.64
	Medicine	99.19	99.73	99.46
	Politics	100	99.73	99.86
	Sports	99.19	99.73	99.46
	Tech	98.64	98.38	98.51
AlKhaleej	Culture	98.13	96.77	97.44
	Finance	99.22	98.00	98.61
	Medicine	98.17	99.08	98.62
	Politics	98.62	99.08	98.85
	Religion	97.41	98.46	97.93
	Sports	99.69	99.69	99.69
	Tech	98.45	97.85	98.15
Akhbarona	Culture	94.29	96.12	95.20
	Finance	91.78	93.28	92.52
	Medicine	95.38	98.66	96.99
	Politics	93.91	89.70	91.76
	Religion	98.65	98.51	98.58
	Sports	99.55	98.21	98.87
	Tech	97.99	94.63	96.28
SANAD	Culture	95.82	95.61	95.71
	Finance	94.70	96.15	95.42
	Medicine	97.60	98.82	98.21
	Politics	97.79	94.14	95.93
	Religion	96.64	97.95	97.29
	Sports	99.52	99.05	99.29
	Tech	97.34	97.28	97.31

4.1 Ablation Analysis

An ablation study was performed by systematically removing some components to assess the efficacy of the proposed architecture. In the follow-up, we review and critically compare model setting and model performance data with the goal of separating the impact of BiLSTM, CNN and Capsule Network layers. Table 5 summarizes the ablation study considering the influence of different components of the model on performance over datasets. The models are:

- AraBERT (Baseline): This is also used as the baseline of our method. The system established a benchmark accuracy of 94.19% on the aggregated dataset; it received scores of 91.75%, 96.81% and 95.91% on the Akhbarona, AlArabiya and AlKhaleej sub-sets, respectively. All subsequent models are evaluated against this performance.
- AraBERT + BiLSTM: The addition of a BiLSTM layer yielded significant improvements across the board. On the SANAD, the accuracy increased by 1.45% to 95.64%. The most substantial impact was observed on the Akhbarona sub-set, where performance climbed by 2.88% from 91.75% to

94.63%. This confirms the value of modeling sequential context, especially for more complex datasets.

- AraBERT + CNN: Augmenting the baseline with CNN layers resulted in the most significant performance gain from a single component. It increased the SANAD score by 2.14% to 96.33%. Notably, its performance on AlArabiya (98.70%) and AlKhaleej (98.00%) demonstrates the strength of CNNs in extracting highly informative local features.
- AraBERT + CNN + Capsule: The model that incorporates the AraBERT model with CNN and then adds the Capsule layer achieved an accuracy of 95.93% on the aggregated dataset. What is interesting to note is that this model achieved an accuracy that was 0.40 percentage points lower than the AraBERT-CNN model. This result reveals an important architectural insight: the capsule-projection layer is a relational feature-fusion mechanism the effectiveness of which is conditioned on the structural richness of its input. Without BiLSTM, Global Max Pooling reduces the CNN output to a bag of independently selected scalar activations, discarding all sequential and relational structures; projecting such a vector through a capsule layer cannot recover this discarded information and the additional parameters introduce optimization noise that marginally degrades performance. In contrast, when BiLSTM is present in the full ABPC-Net pipeline, the capsule layer receives structurally richer input and its contribution becomes clearly positive: ABPC-Net (97.02%) outperforms AraBERT + BiLSTM + CNN (95.89%) by 1.13% on SANAD, confirming that the capsule projection effectively encodes relational feature structure when provided with sufficiently rich sequential input.
- AraBERT + BiLSTM + CNN: This model was a direct integration of components in our analysis. It reported 95.89% accuracy on the SANAD dataset. This, importantly, is 0.44% less than that achieved by the AraBERT + CNN model. The same performance degradation was observed in other sets, such as AlArabiya (0.75% drop) and AlKhaleej (0.42% drop). This is the quantitative evidence that the naive combination of these layers is non-optimal and doesn't properly harmonize the features being extracted.
- ABPC-Net: Finally, in this part, our proposed architecture alleviated degradation observed in the previous two architectures. In the case of the SANAD dataset, an accuracy of 97.02% was obtained by a cleverly-integrated parallel feature output from the BiLSTM and the CNN layers *via* the Capsule Network. It indicates 0.69% improvement over the best single-component model and 1.09% over the AraBERT + CNN + Capsule model. This model displayed the best performance in all the individual sub-sets.

Table 5. Ablation analysis on AlArabiya, AlKhaleej, Akhbarona and SANAD datasets.

Methods	AlArabiya	AlKhaleej	Akhbarona	SANAD
AraBERT	96.81	95.91	91.75	94.19
AraBERT + BiLSTM	98.00	96.88	94.63	95.64
AraBERT + CNN	98.70	98.00	95.14	96.33
AraBERT + CNN + Capsule	98.32	97.41	95.01	95.93
AraBERT + BiLSTM + CNN	97.95	97.58	94.88	95.89
ABPC-Net	99.14	98.42	95.59	97.02

In summary, this numerical analysis confirms that while BiLSTM and CNN operate effectively as feature-extraction components, their successful integration depends on the architectural context in which they are combined. In particular, the capsule-inspired projection is not simply an additional component, but acts as a fusion mechanism that enables structured integration of parallel feature streams. The comparison between AraBERT + BiLSTM + CNN (95.89%) and ABPC-Net (97.02%) provides an estimate of the effect of introducing a vector-based projection head between feature aggregation and classification. The observed improvement of approximately 1.13% on SANAD, along with consistent gains across sub-sets, indicates a representational benefit over scalar classification in this specific architectural setting. Furthermore, the vector-length-based decision mechanism provides an

interpretable class-activation signal, which may be beneficial in scenarios where feature boundaries are ambiguous. This result highlights that the effectiveness of the capsule-inspired projection is conditional rather than universal. Specifically, its performance gain emerges only when the input representation preserves sequential structure, as provided by the BiLSTM layer. This finding suggests that capsule-inspired mechanisms are more appropriately viewed as relational fusion operators than as standalone classifiers, particularly in Arabic-text classification settings.

4.2 Comparison with Recent Published Methods

The performance advantages of ABPC-Net over recent transformer-based models are attributable not to the transformer backbone itself, which is identical to the AraBERT baseline, but to the structured downstream architecture that better exploits the high-dimensional representations produced by the frozen encoder. To contextualize the performance of ABPC-Net against published Arabic-text classification methods, we present a comparison with a set of recent models on SANAD, AlArabiya, AlKhaleej and Akhbarona datasets in Table 6. We emphasize at the outset that the results in Table 6 for competing methods are reproduced directly from their respective original publications and were not re-implemented by the authors under identical experimental conditions. Although the standard SANAD train/test split is widely adopted across these works, the external results may still differ in pre-processing pipelines, validation protocols, hyper-parameter tuning, framework implementations, hardware environments and in single-run *versus* multi-run reporting. On AlKhaleej sub-set, ABPC-Net reaches an accuracy of $98.40 \pm 0.10\%$, which compares favorably to the published results of CNN with character-level model [14] (98.00%) and Tasneef [11] (97.49%). This result reflects the effectiveness of the hybrid architecture in classifying articles from this news source, with consistent gains over CNN and character-level feature-extraction approaches. Furthermore, a recently proposed model integrating Graph Convolutional Networks (GCNs) with AraBERT embeddings [26] achieves 97.25% on AlKhaleej, representing a qualitatively different architectural direction from sequence-based methods. ABPC-Net shows a margin of 1.15% relative to this graph-based model (98.40% *vs.* 97.25%).

Table 6. Accuracy comparison of the ABPC-Net model against SOTA methods.

Methods	AlArabiya	AlKhaleej	Akhbarona	SANAD
BiGRU [5]	97.41	96.46	92.23	94.83
Attention-GRU [5]	96	96.66	92.95	94.98
CGRU [5]	97.19	96.86	94	95.71
ArCAR [10]	-	97.47	-	-
CNN with character level [16]	-	98	-	-
Transformer-CNN [27]	97.19	96.55	92.14	94.29
Tasneef [11]	98.43	97.49	95.43	-
TCAODL-ANA [28]	-	-	-	95.48
Inception-CNN + LSTM [4]	82	96	92	92
GCN+AraBERT [26]	-	97.25	-	-
ABPC-Net	$\mathbf{99.14 \pm 0.10}$	$\mathbf{98.40 \pm 0.10}$	$\mathbf{95.59 \pm 0.12}$	$\mathbf{97.00 \pm 0.04}$

*All values are reported in percentages (%).
**The bold values indicate the high-accuracy performance achieved in each comparison.
*** Results for external models are sourced from original publications and may reflect different pre-processing pipelines, train/test splits or hardware environments.

For AlArabiya dataset, our model achieves strong performance in classification with an accuracy of 99.14%. That's 0.71% better than Tasneef's [11] previous best result (98.43%). Thus, this indicates a level of accuracy; it demonstrates that the ABPC-Net method has learned the linguistic structures of AlArabiya dataset, indicating that contextual-local-hierarchical feature combinations have proven effective. On the more challenging Akhbarona sub-set, ABPC-Net achieves an accuracy of $95.59 \pm 0.12\%$, which compares favorably to the published results of Tasneef [11] (95.43%), CGRU [5]

(94.00%) and Transformer-CNN [27] (92.14%). The consistency of these numerical advantages across multiple comparison points suggests that the architecture handles Akhbarona's linguistic variability effectively. Finally, on the aggregated SANAD corpus, ABPC-Net achieves an accuracy of $97.00 \pm 0.04\%$, which compares favorably to the published results of CGRU [5] (95.71%) and TCAODL-ANA [28] (95.48%), showing a margin of approximately 1.31% relative to the closest competing entry in this comparison. Taken together, these results position ABPC-Net as a strong-performing approach for Arabic news classification on the SANAD benchmark, though generalization to other datasets and domains warrants further investigation.

4.3 Intra-domain Transferability and Few-shot Adaptation

To assess the intra-domain transferability of ABPC-Net beyond the SANAD benchmark, we conducted a two-phase cross-dataset evaluation using BBC Arabic and CNN Arabic [29], two widely-used Arabic news corpora that, while distinct from SANAD in source, writing style and category structure, remain within the news domain. Accordingly, the experiments in this section evaluate robustness to source-level distribution shift.

Phase 1: Zero-shot Cross-dataset Evaluation

The ABPC-Net model trained exclusively on SANAD was evaluated directly on BBC Arabic (7 categories) and CNN Arabic (6 categories) without any additional training. Since both datasets do not contain the Medical and Religion categories present in SANAD, a systematic category mapping was applied to align label spaces across datasets. Specifically, BBC Arabic categories were mapped as follows: اقتصاد و اعمال→ Finance, العالم → and Politics. CNN Arabic categories were mapped as follows: business → Finance, علوم وتكنولوجيا → Tech, رياضة → Sports, منوعات and عرض الصحف→ Culture, اخبار العالم and اخبار الشرق الاوسط → Politics. It is worth noting that the merging of world and middle_east into a single Politics class, as well as منوعات and عرض الصحف into Culture, introduces label-space asymmetry that partially accounts for the performance variation observed across categories in the zero-shot evaluation. The category mapping applied above introduces three sources of bias that should be considered when interpreting the zero-shot results. First, merging multiple BBC and CNN categories into single SANAD classes creates merged classes that are broader than their SANAD counterparts, which may affect per-class metrics. Second, the merged classes are semantically broader than the original SANAD definitions, which may shift accuracy in either direction depending on alignment with the model's learned class boundaries. Third, both BBC Arabic and CNN Arabic lack the Medical and Religion categories present in SANAD, producing labelspace asymmetry that the zero-shot evaluation cannot fully resolve. Consequently, the reported zero-shot accuracies should be interpreted as they are partially confounded by these label-mapping effects.

Under zero-shot conditions, ABPC-Net achieved 60.76% on BBC Arabic and 75.33% on CNN Arabic. These results reflect the combined effect of source-level distribution shift between news outlets and the label-mapping bias described above. The substantial gap relative to in-domain SANAD performance should therefore be attributed to both factors and the absolute zero-shot values should not be treated as direct measures of cross-source generalization. Importantly, categories with universal linguistic patterns transferred well, Sports achieved 96.99% F1 on CNN Arabic and Politics achieved 83.50% F1, while domain-specific categories, such as Tech (39.31% F1)و showed greater sensitivity to cross-source variation, a finding consistent with cross-domain transfer literature in Arabic NLP. The observed performance gap is further attributable to the absence of Medical and Religion categories in the target datasets, which introduces systematic label-space mismatch.

Phase 2: Few-shot Domain Adaptation

To evaluate adaptability under low-resource conditions, ABPC-Net was fine-tuned using only 20% of BBC Arabic and CNN Arabic, respectively, with the remaining 80% reserved for testing. This protocol simulates a realistic deployment scenario where limited target-domain supervision is available. Following adaptation with only 20% of target domain data, ABPC-Net achieved 94.36% on BBC Arabic and 89.20% on CNN Arabic, demonstrating substantial performance recovery with minimal additional supervision. These results confirm that the ABPC-Net architecture, with its frozen AraBERT encoder and trainable downstream layers, is particularly well-suited for rapid domain adaptation, as only the lightweight BiLSTM-CNN-Capsule pipeline requires updating. Notably, the few-shot adaptation

protocol yielded a performance gain of 33.60 percentage points on BBC Arabic (from 60.76% to 94.36%) and 13.87 percentage points on CNN Arabic (from 75.33% to 89.20%), demonstrating that ABPC-Net's frozen AraBERT encoder retains transferable linguistic representations while the trainable downstream layers, BiLSTM, CNN and Capsule, adapt rapidly to the target domain distribution with minimal supervision. Taken together, these results indicate that ABPC-Net's generalization is bounded, but adaptable: zero-shot transfer is constrained by source-level distribution shift and label-space asymmetry, while few-shot finetuning efficiently bridges these gaps, recovering the majority of source-specific performance using only 20% of target-domain data.

Table 7. Cross-dataset generalization results on BBC Arabic and CNN Arabic.

Evaluation Protocol	Dataset	Accuracy	Training Data
Zero-shot Transfer	BBC Arabic	60.76	SANAD only
Zero-shot Transfer	CNN Arabic	75.33	SANAD only
Few-shot Adaptation	BBC Arabic	94.36	SANAD + 20% BBC
Few-shot Adaptation	CNN Arabic	89.20	SANAD + 20% CNN
Independent Training	BBC Arabic	99.37	BBC only
Independent Training	CNN Arabic	94.68	CNN only

*All values are reported in percentages (%).
** Independent Training results use 70%/10%/20% train/validation/test split.
*** Medical and Religion categories absent in BBC and CNN datasets.

The architecture therefore exhibits consistent intra-domain transferability across Arabic news sources and rapid few-shot adaptability under low-resource conditions. We emphasize, however, that these findings do not establish cross-domain generalization beyond the news genre; evaluation on dialectal Arabic, conversational text, scientific Arabic and other non-news domains remains an important direction for future work and is identified explicitly in the Conclusion. Table 7 summarizes the cross-dataset generalization results.

4.4 Error Analysis

The overall result of the text-classification model performance was highly accurate and the misclassifications were due to some semantic overlap of certain categories, as seen from the text-training dataset. This work is based on the confusion matrix illustrated by Figure 3, which demonstrates the proportion of correct and incorrect predictions. For AlArabiya dataset, as shown in Figure 3(a), a clustering of mistakes appears in the Tech category, which had 2 misclassifications as Finance, 2 as Medical and 2 as Sports, according to the confusion matrix. Similarly, the Finance category had a misclassification of the same case as Tech, only 1. While indicating high overall accuracy, this highlights that there is a slight overlap between the categories with similar terminology.

In AlKhaleej dataset, as shown in Figure 3(b), the confusion matrix reveals a specific focus on errors between the Finance and Tech categories. Specifically, 8 Finance articles were misclassified as Tech, while 4 Tech articles were classified as Finance. A clear example of this ambiguity is found in an article stating:"
استحوذت ميديا كويست، الرائدة في مجال الاعلام في منطقة الشرق الأوسط وشمال إفريقيا، على ما قدره 30% من أسهم الموقع الإلكتروني Whiteme.net."

This article was misclassified as Tech instead of Finance. The likely reason for this error is the dominance of technical terms such as "المواقع الإلكترونية" (websites) and the specific domain name "Whiteme.net". The model appears to have prioritized these technical features over the broader financial context established by key terms, like "أسهم" (stocks) and "استحوذت" (acquired).

An example is an article titled "حل جديد شحن اجهزه الكترونيه طاقه شمسيه" (discussing innovative solar-powered charging solutions for electronic devices), which was predicted as Tech instead of Finance. The text emphasizes technical details, such as "طاقه شمسيه" (solar energy), "شاحن" (charger) and "بطاريه ليثيوم" (lithium battery), which likely dominated the convolutional filters and aligned closely with Tech

vocabulary. The model's reliance on these technology-centric n-grams overshadowed subtle financial undertones, like market applications, leading the capsule projection mechanism to favor Tech Class.

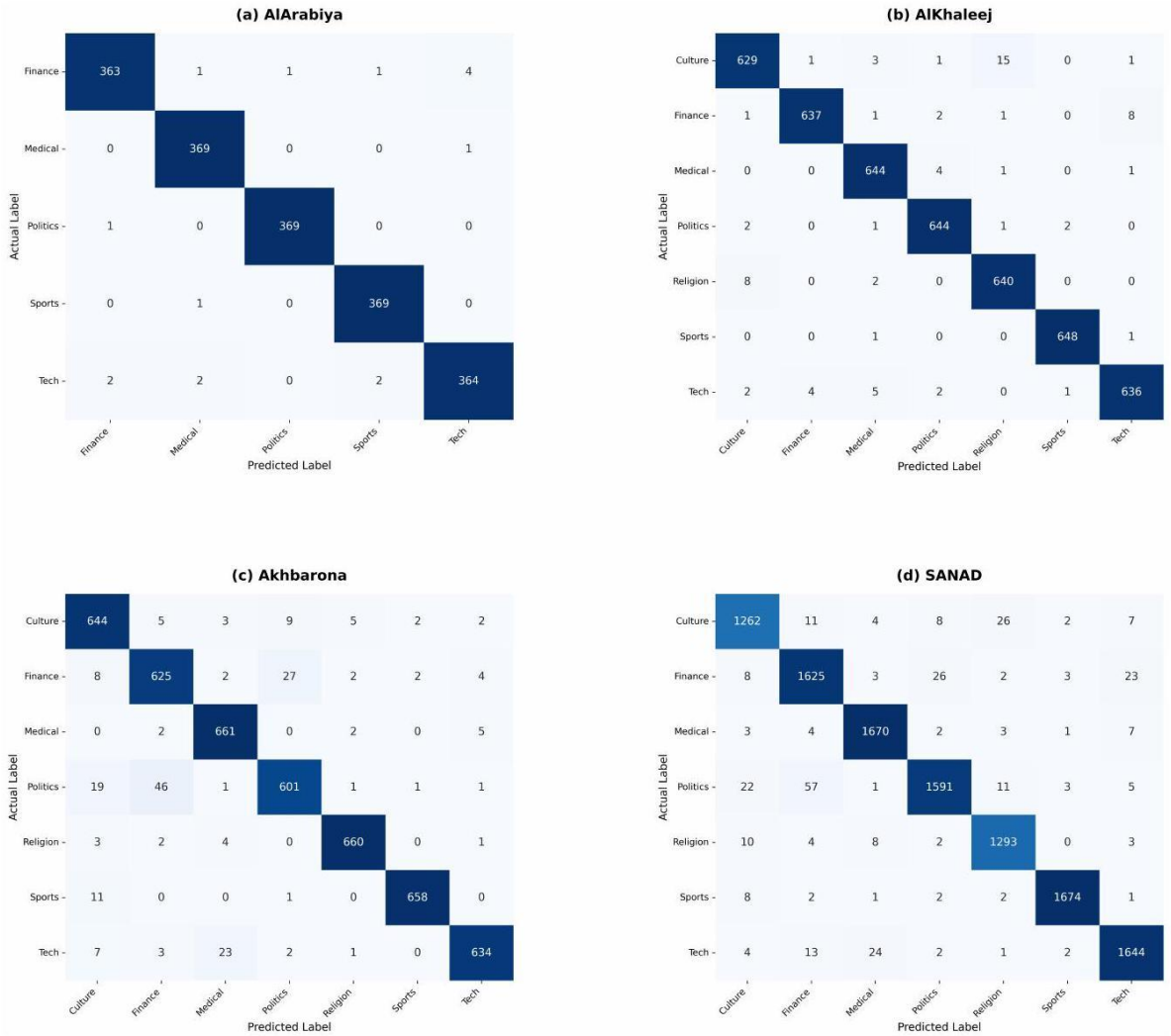

Figure 3. Confusion matrices for AlArabiya (a), AlKhaleej (b), Akhbarona (c) and SANAD (d) datasets, illustrating per-class classification performance and primary sources of misclassification.

As for Akhbarona dataset, Figure 3(c), the confusion matrix highlights prominent errors in the Politics category, with 46 cases misclassified as Finance and 19 as Culture, indicating strong semantic overlap between these domains. For example, an article titled: "ميزانية وزارة الاوقاف ارتفعت بأزيد من 2000 بالمائة", was misclassified as Finance instead of Politics. This error is attributed to the presence of strong financial terms, like "ميزانية" (Budget) in the title, along with "الاستثمار" (Investment) and "السنة المالية" (Fiscal year) in the body text, which outweighed the political context of the ministry's activities in the model's decision-making process.

Finally, for the aggregated SANAD dataset, Figure 3(d), the confusion matrix indicates that the combination of data increased the complexity of the data. Among the many examples, there is a notable focus on misclassification of Politics as Finance with 57 cases of Politics classified as Finance. This illustrates the complexity of reconciling differing contexts when combining diverse sources. Data integration and better generalization were achieved, but also exacerbated specific ambiguities, such as lingering Tech and Finance exchanges.

4.5 Limitations and Dataset Bias

Although ABPC-Net achieves strong empirical performance on the evaluated benchmarks, several important limitations should be acknowledged for accurate interpretation of the results. These limitations relate primarily to dataset bias and to the scope of the experimental evaluation.

First, all evaluated datasets, SANAD, BBC Arabic and CNN Arabic, consist exclusively of Modern Standard Arabic (MSA) news text. However, Arabic encompasses a wide spectrum of dialects and registers, including Egyptian, Levantine, Gulf and Maghrebi varieties, as well as informal written forms commonly used in social media and conversational platforms. These variants differ substantially from MSA in morphology, syntax and lexical usage. ABPC-Net has not been evaluated on such dialectal or informal data and its performance in these settings remains an open question.

Second, because ABPC-Net employs a frozen AraBERT encoder, it inherits the pre-training distribution of AraBERT, which is predominantly based on MSA corpora. As a result, the effectiveness of the proposed downstream architecture is closely related to this representation space. Extending the approach to dialectal Arabic would likely require the use of dialect-aware or multi-dialect pre-trained encoders.

Third, all evaluated datasets are drawn from the news domain, which tends to emphasize political, financial and sports content while under-representing other genres, such as scientific, technical, legal and conversational text. Consequently, the reported performance should be interpreted within the scope of Arabic news classification rather than as a general indicator of performance across all Arabic-text domains.

Fourth, SANAD dataset itself reflects specific editorial styles associated with Gulf and pan-Arab news portals (AlArabiya, AlKhaleej and Akhbarona), which differ from other regional styles. This contributes to the source-level distribution shift observed in the cross-dataset experiments reported in Sub-section 4.3.

Fifth, the observed drop in zero-shot performance between SANAD and external datasets (BBC Arabic and CNN Arabic) further highlights the sensitivity of the model to distribution shift, even within the news domain. While few-shot adaptation substantially improves performance under limited supervision, these results suggest that careful adaptation remains important when transferring to new sources.

Sixth, the ablation study compares the capsule-inspired projection against a scalar baseline (AraBERT+BiLSTM + CNN with direct margin loss) but does not exhaustively compare against alternative post-pooling mechanisms, such as deeper Dense heads, gating or branch-level attention. Whether similar gains could be achieved using lower-complexity alternatives remains an open question.

Overall, these limitations indicate that the reported results demonstrate strong performance within the scope of MSA Arabic news classification, while broader evaluation across dialects, domains and genres, as well as direct comparison against simpler architectural alternatives, remains an important direction for future work.

5. CONCLUSION

In this article, we have presented ABPC-Net as a structured downstream architectural design for transformer-based Arabic-text classification. The contribution is primarily architectural and empirical, supported by systematic experimental analysis. Their interaction yields non-trivial performance gains and provides practical insights into effective downstream design - particularly regarding the conditional effectiveness of capsule-inspired projections, which function as relational fusion operators rather than standalone classifiers. The capsule-inspired projection contributes consistent performance improvements when used within an appropriate architectural context, highlighting its role as a complementary component rather than a standalone solution.

We presented a rigorous ablation analysis showing that the fusion of these aspects is a key factor to performance. When combined, the ABPC-Net model demonstrates consistent improvements over individual baselines and hybrid configurations under the evaluated settings on the SANAD benchmark. Extensive evaluation yields a mean accuracy of $97.00 \pm 0.04\%$ on the full SANAD dataset, with particularly strong performance on AlArabiya ($99.14 \pm 0.10\%$), though these results should be interpreted within the scope of the tested datasets and experimental conditions. Furthermore, cross-dataset evaluation on BBC Arabic and CNN Arabic provides evidence of consistent intra-domain transferability across Arabic news sources and of rapid few-shot adaptability under low-resource conditions.

However, in spite of these encouraging findings, there were some limitations and drawbacks in our study. One such difficulty is the computational complexity induced by the hybrid structure, hierarchical

feature representation enabled by the capsule-inspired vector encoding mechanism, which delays the training of the model. On closer inspection, misclassification patterns persisted between closely related categories, such as Politics and Finance, indicating that contextual overlap remains challenging even for sophisticated architectures. Several limitations of the current work warrant acknowledgment. First, SANAD comprises exclusively Modern Standard Arabic news articles drawn from three specific portals, introducing source-specific stylistic and topical biases that may limit generalization to other domains or writing styles. The cross-dataset evaluation on BBC Arabic and CNN Arabic (Sub-section 4.3) provides empirical evidence of this constraint, where zero-shot transfer performance reflects the domain shift between the training distribution and unseen target sources. Second, as AraBERT is pre-trained predominantly on MSA corpora, ABPC-Net inherits an inherent limitation in handling dialectal Arabic variants, including Egyptian, Levantine and Gulf dialects, which differ substantially from MSA in morphology, vocabulary and syntactic structure. Extending the architecture to dialectal Arabic through dialect-aware pre-trained encoders or multi-dialect training data remains an important direction for future work.

In the future, we want to investigate the model compression based on knowledge distillation or quantization methods, thereby aiming to reduce computation demand and inference time without decreasing accuracy and making the model more practical in actual applications. In order to mitigate this issue of semantic overlap, we intend to explore more complex types of data augmentation, such as back-translation or contextual word replacement, to generate more diverse training samples to solve the problem of class pairs. Additionally, we extend the architecture to support dialectal Arabic variations (e.g., Egyptian, Levantine) and apply XAI methods (explainable AI) to visualize the routing decision of the Capsule Network, so that we might achieve greater transparency and wider impact on Arabic NLP tasks. A promising direction for future work is replacing the BiLSTM layer with a lighter temporal convolutional network (TCN) for sequence modeling, which may achieve similar representational capacity with lower computational cost. This would further explore the trade-off between sequential-modeling effectiveness and efficiency in transformer-based hybrid architectures.

REFERENCES

[1] A. Alrayzah, F. Alsolami and M. Saleh, "AraFastQA: A Transformer Model for Question-answering for Arabic Language Using Few-shot Learning," Computer Speech & Language, vol. 95, p. 101857, 2026.

[2] A. Wali et al., "Evaluating Arabic Sentiment Analysis with GPT-4o: A Comparative Study of Raw and Pre-processed Text," J. of Cases on Inf. Tech., vol. 28, no. 1, pp. 1-19, 2026.

[3] J. H. Yousif, "Artificial Intelligence and Machine Learning for Enhancing Arabic Fake News Detection: A BERT-based Transformer Approach," Procedia Computer Science, vol. 275, pp. 809-816, 2026.

[4] E. Alnagi, R. Ghnemat and Q. Abu Al-Haija, "Boosting Arabic Text Classification Using Hybrid Deep Learning Approach," Discover Applied Sciences, vol. 7, no. 6, p. 540, May 2025.

[5] A. Elnagar, R. Al-Debsi and O. Einea, "Arabic Text Classification Using Deep Learning Models," Information Processing & Management, vol. 57, no. 1, p. 102121, Jan. 2020.

[6] N. Boudad, R. Faizi, R. Oulad Haj Thami and R. Chiheb, "Sentiment Analysis in Arabic: A Review of the Literature," Ain Shams Engineering Journal, vol. 9, no. 4, pp. 2479-2490, Dec. 2018.

[7] B. M. Merzah, J. Razmara and Z. Salmanian, "Hybrid Deep Learning Models for Fake News Detection: Case Study on Arabic and English Languages," Frontiers in Big Data, vol. 8, DOI: 10.3389/fdata.2025.1683786, Jan. 2026.

[8] A. B. Nassif et al., "Arabic Fake News Detection Based on Deep Contextualized Embedding Models," Neural Computing & Applications, vol. 34, no. 18, pp. 16019-16032, Sep. 2022.

[9] I. Guellil, H. Saâdane, F. Azouaou, B. Gueni and D. Nouvel, "Arabic Natural Language Processing: An Overview," J. of King Saud Uni.-Computer and Inf. Sciences, vol. 33, no. 5, pp. 497-507, 2021.

[10] A. Y. Muaad et al., "A Novel Deep Learning ArCAR System for Arabic Text Recognition with Character-level Representation," Computer Sciences and Mathematics Forum, vol. 2, no. 1, Article no. 14, DOI: 10.3390/IOCA2021-10903, and Presented at the 1st Int. Electronic Conf. on Algorithms, Sep. 2021.

[11] M. Louail et al., "Tasneef: A Fast and Effective Hybrid Representation Approach for Arabic Text Classification," IEEE Access, vol. 12, pp. 120804-120826, 2024.

[12] I. Jamaleddyn, R. ayachi and M. Biniz, "Novel Multichannel Deep Learning Model for Arabic News Classification," Jordanian Journal of Computers and Information Technology (JJCIT), vol. 10, no. 4, pp. 453-468, DOI: 10.5455/jjcit.711720086134, Dec. 2024.

[13] Md. M. Hossain et al., "A Hybrid Attention-based Transformer Model for Arabic News Classification Using Text Embedding and Deep Learning," IEEE Access, vol. 12, pp. 198046-198066, 2024.

[14] M. El Kourdi, A. Bensaid and T. E. Rachidi, "Automatic Arabic Document Categorization Based on the Naïve Bayes Algorithm," Proc. of the Workshop on Computational Approaches to Arabic Script-based Languages (Semitic '04), pp. 51-58, Geneva, Switzerland, 2004.

[15] L. Al Qadi, H. El Rifai, S. Obaid and A. Elnagar, "A Scalable Shallow Learning Approach for Tagging Arabic News Articles," Jordanian Journal of Computers and Information Technology (JJCIT), vol. 6, no. 3, pp. 263-280, DOI: 10.5455/jjcit.71-1585409230, 2020.

[16] A. Y. Muaad et al., "An Effective Approach for Arabic Document Classification Using Machine Learning," Global Transitions Proceedings, vol. 3, no. 1, pp. 267-271, Jun. 2022.

[17] D. Alsaleh and S. Larabi-Marie-Sainte, "Arabic Text Classification Using Convolutional Neural Network and Genetic Algorithms," IEEE Access, vol. 9, pp. 91670-91685, 2021.

[18] M. S. H. Ameur, R. Belkebir and A. Guessoum, "Robust Arabic Text Categorization by Combining Convolutional and Recurrent Neural Networks," ACM Trans. on Asian and Low-resource Language Information Processing, vol. 19, no. 5, pp. 1-16, DOI: 10.1145/3390092, Sep. 2020.

[19] A. A. Jalil and A. H. Aliwy, "Classification of Arabic Social Media Texts Based on a Deep Learning Multi Tasks Model," Al-Bahir, vol. 2, no. 2, DOI: 10.55810/2313-0083.1030, May 2023.

[20] B. B. Al-onazi et al., "Automated Arabic Text Classification Using Hyper-parameter Tuned Hybrid Deep Learning Model," Computers, Materials & Continua, vol. 74, no. 3, pp. 5447-5465, 2023.

[21] W. Antoun, F. Baly and H. Hajj, "AraBERT: Transformer-based Model for Arabic Language Understanding," Proc. of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection, pp. 9-15, Marseille, France, 2020.

[22] A. Jalili, H. Tabrizchi, J. Razmara and A. Mosavi, "BiLSTM for Resume Classification," Proc. of the 2024 IEEE 22nd World Symposium on Applied Machine Intelligence and Informatics (SAMI), pp. 519-524, Stará Lesná, Slovakia, 2024.

[23] Y. Kim, "Convolutional Neural Networks for Sentence Classification," Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1746–1751, Doha, Qatar, DOI: 10.3115/v1/D14-1181, 2014.

[24] S. Sabour, N. Frosst and G. E. Hinton, "Dynamic Routing between Capsules," Proc. of the 31st Conf. on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA., vol. 30, 2017.

[25] D. M. W. Powers, "Evaluation: From Precision, Recall and F-measure to ROC, Informedness, Markedness and Correlation," arXiv: 2010.16061, 2020.

[26] M. Benhammouda, A. Khobzaoui and N. Mahammed, "Arabic Text Classification Using Graphs and Deep Learning," IJCESEN, vol. 11, no. 4, pp. 9415-9421, DOI: 10.22399/ijcesen. 4402, 2025.

[27] M. Berrimi, M. Oussalah, A. Moussaoui and M. Saidi, "A Comparative Study of Effective Approaches for Arabic Text Classification," Available at SSRN 4361591, DOI: 10.2139/ssrn. 4361591, Feb. 2023.

[28] M. S. A. Alzaidi et al., "Enhanced Automated Text Categorization *via* Aquila Optimizer with Deep Learning for Arabic News Articles," Ain Shams Engineering Journal, vol. 16, no. 1, p. 103189, DOI: 10.1016/j.asej.2024.103189, Jan. 2025.

[29] M. K. Saad and W. Ashour, "OSAC: Open Source Arabic Corpora," Proc. of the 6th ArchEng Int. Symposiums on Electrical and Electronics Engineering and Computer Science (EEECS'10), vol. 10, p. 55, DOI: 10.13140/2.1.4664.9288, 2010.

Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 12, No. 02, June 2026.

ملخص البحث:

لا يزال تصنيف النّصوص العربية يمثل تحدياً مهماً بسبب الثراء الصرفي والتّنوع اللّهجي والتّعقيد الدّلالي الّذي تتميز به اللّغة العربية. وعلى الرغم من التّقدم الذي حقّقته النّماذج المعتمدة على المحولات (Transformers) مثل AraBERT، فإن العديد من الأساليب الحالية ما تزال تعتمد على طبقات تصنيف بسيطة لا تستفيد بصورة كاملة من الخصائص الهرمية والتسلسلية للنّصوص.

تقدّم هذه الدّراسة إطاراً هجيناً بإسم ABPC-Net يجمع بين مشفر AraBERT مجمداً لاستخراج التمثيلات السياقية، وشبكة ثنائية الاتجاه من نوع (BiLSTM) لنّمذجة الاعتماديات التسلسلية، وفروعاً متوازية من الشبكات العصبية الالتفافية (CNN) متعددة المقاييس بأحجام نوافذ (2 و 3 و 4) لاستخراج خصائص n-gram، بالإضافة إلى رأس إسقاط متّجهي مستوحى من شبكات الكبسولات (Capsule Networks) لدمج الخصائص الهرمية.

تمّ تقييم النّموذج على مجموعة بيانات SANAD ومجموعاتها الفرعية (العربية، والخليج، وأخبارنا) عبر خمس تجارب مستقلة. وحقّق متوسط دقّة بلغ 97.00±0.04% على SANAD، و99.14±0.10% على العربية، و98.40±0.10% على الخليج، و95.59±0.12% على أخبارنا، متفوقاً بصورة متسقة على نماذج AraBERT و MARBERT المجمدة والمضبوطة بالكامل ضمن الظروف التجريبية نفسها.

كما أظهرت التجارب العابرة لمجموعات البيانات على BBC Arabic وCNN Arabic قدرة النموذج على الانتقال المعرفي داخل مجال الأخبار العربية والتّكيف السريع مع مصادر بيانات جديدة باستخدام عدد محدود من أمثلة التّدريب.

وتشير النّتائج إلى أنّ البنية المقترحة تمثل إطاراً فعالاً وقابلاً للتّعميم لتّصنيف النّصوص الإخبارية العربية المكتوبة باللّغة العربية الفصحى الحديثة.