[1]Amani Aljehani

[2]Syed Hamid Hasan

A BERT-based Prototypical Networks for Few-Shot Arabic Short-text Topic Detection

Abstract: - With the rapid growth of social media platforms, messaging apps, and online forums, vast amounts of short textual content are generated daily in Arabic, covering a wide range of topics and discussions. The ability to automatically detect the topics within these short texts is crucial for various applications. State-of-the-art deep learning models demonstrate high performance in this particular task. However, these approaches may encounter challenges in acquiring knowledge of the semantic space, and their effectiveness significantly depends on the availability of extensive, annotated training datasets. Unfortunately, the Arabic language lacks sufficient resources in this regard. In this paper, we propose a few-shot learning model for Arabic short-text topic detection, where the model proves its ability to generalize from a few examples to new, unseen classes leveraging prior knowledge from related tasks. The model's performance was assessed on three short-text datasets that are publicly available (SemEval, ASND, and AITD). The experimental results demonstrate that the proposed model outperforms baseline models considering only 1, 5, and 10 examples during the training phase, providing empirical evidence for the effectiveness of employing few-shot learning in text classification tasks.

Keywords: Few-shot learning, Arabic short-text, Topic detection, Deep learning

1 INTRODUCTION

Text classification is a fundamental task in NLP which refers to the systematic procedure of assigning a certain class or category to a given text based on its content, utilizing a predetermined set of categories [1]. The objective is to analyze and classify text data automatically in order to facilitate information retrieval, organization, and decision-making. Many fields have used text classification for various purposes. Such as in email spam filtering to differentiate between legitimate and spam messages, and in document classification for tasks such as topic detection, genre classification, and content filtering. Also, it is used to classify news articles into various topics or domains for the news classification application [1]. Several studies have explored traditional machine learning algorithms, such as Support Vector Machines (SVM), Naive Bayes, and Decision Trees for topic detection tasks. These methods often rely on feature engineering techniques and bag-of-words representations to categorize text into predefined topics. Furthermore, probabilistic models like Latent Dirichlet Allocation (LDA) have been extensively employed in capturing topic distributions within large document collections [2]. Such methods, although effective, face challenges in handling short texts, especially in languages like Arabic, where morphological complexities add a layer of difficulty to the analysis. To address this, recent research has focused on leveraging deep learning techniques, including Long Short-Term Memory (LSTM), Convolutional Neural Networks (CNNs), Recurrent Neural Networks, and Transformer-based models, which excel in capturing semantic relationships and contextual information within short texts. These models, often required to be pre-trained on large annotated textual corpora, enable the extraction of intricate patterns and enhance the accuracy of topic detection in diverse linguistic contexts. This presents a challenge in low-resource languages or in domains with limited annotated data such as Arabic language. Arabic is a diverse and rich language that is known for its historical importance, cultural history, and wide use across the Middle East and North Africa. However, it presents some unique challenges for traditional topic detection systems. Since Arabic has a rich morphological structure, with a large number of inflectional and derivational morphemes, which can be attributed to the phenomenon of word derivation from a single root, resulting in the creation of numerous words with distinct meanings [3]. For example, the terms “book” (i.e., “كتاب”), “writing” (i.e., “كتابة”), “writer” (i.e., “كاتب”), and “library” (i.e., “مكتبة”) are all derived from the same linguistic root (i.e., “كتب”) which can affect the accuracy of stemming algorithms [4]. Additionally, Arabic language is written from right to left, which can pose challenges for text alignment and feature extraction, and the way its letters are put together is very complicated. Each letter in Arabic can look different based on where it is in a word.

[1,2] Department of Information Systems, Faculty of Computing and Information Technology, King AbdulAziz University.

email: [1]aaljehani0217@kau.edu.sa,[2]shhasan@kau.edu.sa

Moreover, diacritic marks are applied to or positioned around letters in order to indicate the pronunciation and so enhance the comprehensibility of words [5]. For example, the word "write" (i.e.,"كَتَبَ") differs from the word "books" (i.e.,"كُتُبْ"). However, the majority of current Arabic writings are composed without the inclusion of diacritical marks. Furthermore, the Arabic language exhibits a distinction between the formal written form, known as Classical Arabic or Modern Standard Arabic (MSA), and the various regional dialects that are spoken in different Arabic-speaking nations, Dialectal Arabic (DA) [6]. Besides, the Arabic short text poses a distinct difficulty of its own due to its sparseness, rare word encounters, and lack of contextual meanings, whereas traditional word embedding models exhibit lower performance when applied to such data resources.

Researchers are exploring methods to address these challenges, such as the use of Arabic-specific preprocessing techniques that can handle the complex morphology and word order of Arabic language [7]. However, these machine/deep learning models often require substantial labeled data for acquiring knowledge of the semantic space. Whereas, in real-world scenarios, obtaining labeled data for every possible topic is a challenging and time-consuming task [8].

In response to these challenges, few-shot learning (FSL) techniques have emerged as powerful solutions, enabling models to generalize and recognize patterns from a few examples. Few-shot learning focuses on training machine/deep learning models to understand and classify new, unseen classes with only a handful of labeled examples [9]. This paradigm shift seems as a viable option for the rapid adaptation to dynamic online content, where new topics emerge frequently, and traditional models struggle to keep up.

In this study, we were motivated by FSL to explore the effectiveness of applying it in Arabic short-text topic detection tasks, and due to the insufficient availability of a substantial number of examples for training classifiers. Our goal is to develop a robust and efficient model capable of accurately classifying topics within short Arabic texts, even with limited labeled examples. By harnessing the power of few-shot learning, we aim to enhance the adaptability and responsiveness of topic detection systems in the ever-evolving Arabic digital environment.

The following is a list of our research's contributions:

- We propose a few-shot topic detection model to deal with Arabic short text. To the best of our knowledge, our model is the first that performs topic detection in Arabic language utilizing a few examples during the training phase.
- The proposed model utilizes a transformer BERT-based model (MARBERT), which provides efficient extraction of Arabic language contextualized embeddings and intricate pattern recognition. Combined with the Prototypical meta-learning framework that allows the model to generalize well to recognize new unseen classes.
- The approach proposed in our study demonstrates superior performance compared to the existing state-of-the-art models across three distinct Arabic short-text datasets.

The rest of the paper is organized as follows: It begins with a review of related work (Section 2) and provides foundational knowledge in Section 3. The core methodology, integrating MARBERT and Prototypical Networks, is detailed in Section 4, followed by the experimental setup and evaluation metrics in Section 5. Section 6 presents the experimental results and in-depth analysis. The paper concludes in Section 7, summarizing contributions and proposing future research directions.

2 RELATED WORK

This section provides a comprehensive review of the existing literature on topic detection tasks in Arabic language. In addition to few-shot learning, including its definition, possible methodologies for implementation, and relevant studies that have used this methodology in text classification problems.

2.1 Topic detection

Topic detection (TD) is the process of categorizing text based on its content, and assigning it to one or more predefined categories. The goal is to automatically identify and extract the main topics discussed in a given text or document, allowing for effective textual data organization, retrieval, and analysis [10]. Reviewing the

literature, numerous methodologies have been discovered that effectively facilitate the identification of topics through the utilization of SOTA machine and deep learning approaches. [11] propose a method named High Utility Pattern Mining (HUPM) that considers both frequency and utility techniques, to identify trending topics on Twitter. They define the utility of terms based on the growth rate in frequency and use HUPM to uncover clusters of phrases that are both often used and highly useful for a subset of tweets extracted from the Twitter stream using a time-based windowing technique. An effective data structure called a Topic-tree is also presented for use in post-processing to extract real topic patterns from the candidate topic patterns created by HUPM. [12] Propose a novel text-mining research framework using Neural Topic Embedding (NTE). This is the first technique that used NTE because it is capable of generating usable and interpretable representations of texts using deep neural networks. Specifically, they use topic modeling to provide meaning to deep-learning data representations. They conducted a preliminary assessment experiment on a testbed of fake review identification to illustrate the usefulness of their suggested framework, and their interpretable representations enhance the state-of-the-art by over 8% as measured by the F1 score. [13] Used Transformers in conjunction with an incremental community discovery technique to classify sports events. Their suggested model consists of many modules, including BERT, graph methods, and a multimodal named entity recognizer. This combination as a unique approach was termed a memory graph that leverages cognitive memorization in the human brain. When dealing with large amounts of social data, the modularity of their work makes it more applicable in real-world and corporate settings.

[14] Proposed a Twitter topics detection system using Semantic Deep Forest (SDF), a topic categorization technique that combines contextual Word2vec, WordNet, and Deep Forest. In addition, they performed in-depth parameter sensitivity analysis to optimize the SDF parameters for their Tweet topic categorization job.

On the other hand, Arabic topic detection plays a vital role in understanding textual content in the vast and diverse Arabic language domain. Researchers have explored various techniques and methodologies to tackle the challenges posed by Arabic texts, including morphological complexity, dialectal variations, and limited labeled datasets. In recent years, several studies have contributed significantly to advancing the field of Arabic topic detection.

[15] The authors conducted a comparative study on the categorization of Arabic text. A dataset including 2700 Arabic articles was acquired, encompassing a diverse range of classes, each representing a distinct article type. This study included five conventional text classification algorithms. Additionally, the texts were subjected to preprocessing procedures involving stemming and cleaning methodologies. The findings of the study indicate that the Support Vector Machine (SVM) classifier had superior performance compared to the other classifiers.

In [16] the researchers opted to employ word and document embedding techniques as opposed to depending on pre-processing and word-counting representations for the purpose of identifying Arabic articles. The research demonstrated that the utilization of Doc2Vec for the purpose of learning and incorporating word vectors resulted in superior performance compared to conventional text pre-processing techniques. [17] Proposed a methodology for classifying Arabic tweets by employing ensemble learning techniques. The dataset includes 500 tweets, evenly distributed among five distinct categories. The classification process was executed with three classifiers: Sequential Minimal Optimization (SMO), Naive Bayes (NB), and J48. The researchers employed individual classifiers, in addition to ensemble learning techniques. Three approaches, namely bagging, boosting, and stacking, were employed for ensemble learning in each classifier. The findings indicated that the utilization of ensemble methods resulted in enhanced accuracy for each individual classifier.

[18] Conducted research on the classification of Arabic tweets utilizing two distinct deep learning methods, namely CNN and RNN. The researchers obtained a dataset of 160,870 Arabic tweets by utilizing the Twitter API. The dataset has been categorized into eight distinct classes. The performance of the deep learning models exhibited minimal differences. The researchers documented an accuracy rate of 90.1%. Additionally, they observed macro-F1 performance scores of 92.71%, 92.86%, and 92.95% while employing CNN, RNN-LSTM, and RNN-GRU models, respectively. [19] Performed an evaluation on the classification of Arabic titles of thesis and dissertations. They employed three commonly used Naive Bayes (NB) classifiers, namely Multinomial Naive Bayes (MNB), Gaussian Naive Bayes (GNB), and

Complemented Naive Bayes (CNB). The writers have gathered a total of 7500 titles encompassing ten distinct areas of study. The researchers documented their highest achievement as an F1 score of 84% while employing the CNB classifier. The accuracy rates for the remaining two classifiers were 81% for Multinomial Naive Bayes (MNB) and 76% for Gaussian Naive Bayes (GNB). All three classifiers achieved comparable performance in both the Geography and Legislation tasks, with an F1 score of 90%. The performance of the MNB classifier was subpar in the domain of linguistic analysis, as it achieved an F1 score below 40%. [10] Investigate the effect of a feature-selection method and an artificial summarization method for topic detection on Arabic articles by employing Vector Space Model (VSM). According to their findings, the system performs better than other automated summarization to cut down on distracting details and improve topic recognition. [20] Proposed an effective approach for categorizing online Arabic news into its appropriate topic. The suggested system employs a variety of natural language processing techniques as well as numerous categorization approaches. The experimental findings reveal that using Information Gain (IG) as a feature selection strategy in conjunction with the Naive Bayes algorithm delivers the highest accuracy in solving the topic recognition issue for online Arabic news. [7] Employs a discriminative multi-nominal Naive Bayes (DMNB) classifier and frequency transform to present an effective strategy for Arabic text classification and topic detection. The proposed method consists of three major steps: Arabic text preprocessing, feature extraction and normalization, and classification. They employ a dataset consisting of 1500 Arabic documents drawn from the Arabic articles corpus and arranged into 5 distinct categories to test the effectiveness of the proposed method. The findings demonstrate that the proposed approach outperforms state-of-the-art approaches in a comparable manner. [21] Aim to test BERTopic with a variety of Arabic language model embeddings and evaluate its performance in comparison to LDA and Non-Negative Matrix Factorization (NMF). To test the efficiency of topic modeling, they employed the Normalized Pointwise Mutual Information (NPMI) metric. When compared to NMF and LDA, BERTopic produced superior outcomes overall. Table 1 summarizes these relevant studies on Arabic topic detection. The majority of prior research has been concentrated on feature extraction methodologies or the development of classifiers that integrate multiple models, leveraging extensively annotated training datasets. This poses a difficulty in languages with limited resources such as Arabic language, or in fields with a scarcity of annotated data. Unlike previously reviewed works, in this study, we approach the Arabic topic detection problem from a novel standpoint and present a model that exhibits high accuracy in categorizing topics within short Arabic texts, even in the absence of sufficient labelled samples. Our proposed model achieves state-of-the-art performance on three distinct datasets.

Table 1 Summary of Related Work for Arabic Topic Detection

Reference	Classification Model/Method	Dataset characteristics		
		Name	Type	# Topics
[10]	VSM	Collected from the Arabic online newspaper Al-Wattan	Documents	6
[15]	SVM, KNN, NB, DT and Decision Table	'Arabic articles' collected by Diab Abu Aiadh	Documents	9
[16]	Doc2Vec with SVM	OSAC	Documents	10
[17]	SMO, NB, J48	Manually collected	Tweets	5
[20]	IG and NB	OSAC	Documents	10
[7]	DMNB	'Arabic articles' corpus collected by Diab Abu Aiadh	Documents	5
[18]	CNN, RNN	Manually collected	Tweets	8
[19]	MNB,GNB,CNB	Manually collected	Thesis titles	10
[21]	BERTopic	DSAC	Documents	5

2.2 Few-shot learning

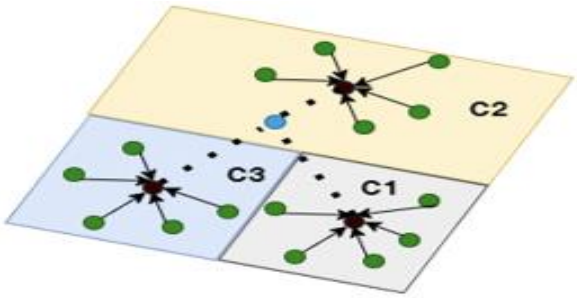
Few-shot learning (FSL) is a machine learning technique that aims to resolve classification or regression problems with a small number of training samples. Unlike standard supervised learning, which requires a large quantity of labelled data to train a model, few-shot learning focuses on developing models that can generalize and forecast based on a few or even one training sample [9]. The goal of few-shot learning is not to teach a model to assemble individual samples from a training set, but rather to teach the model to "learn to learn" and recognize similarities and differences across inputs [9]. The training data in a few-shot learning scenario does not have to comprise samples from all potential prediction classes. A model, for example, can be trained on images of tigers, bunnies, and monkeys before being evaluated on images of cats that were not seen during training. Despite the fact that the model was not specifically trained on cats, it can recognize that these new photos are comparable and belong to the same category based on the learned concept of recognizing similarities and differences [22]. A support set and a query set are used in the few-shot learning setting during the meta-training and meta-testing phases. In meta-testing the support set is made up of a limited number of classes, each with few samples. Even though these classes were not part of the training set, this support set gives additional information to the model during the testing phase. The query set contains examples from previously unknown classes, which the model compares to the support set to detect unseen class labels [22]. Terminologies used in few-shot learning include the *"k-way n-shot"*, where *"k"* represents the number of classes in the support set and *"n"* represents the number of samples per class. For instance, a *3-way 2-shot* support set would have two examples from each of the three categories. The accuracy of the model's predictions is sensitive to the values of *k* and *n*. The model learns a similarity function between query samples and support set samples from several instances within a few sets of related classes, therefore increasing *"n"* may enhance accuracy while increasing *"k"* may lower it [23].

Few-shot learning approaches are classified into various types including the following:

- **The transfer learning approach:** transfer learning is a well-known technique in few-shot learning that employs knowledge from a source domain with abundant labelled data to enhance the performance of a target domain with limited labelled data. In the context of few-shot learning, transfer learning entails pre-training a model on a large dataset and then fine-tuning it using the limited labelled data available for the few-shot task. By leveraging pre-trained knowledge, the model is able to capture general features and representations that are applicable across tasks and domains, thereby enhancing its ability to generalize to new classes or samples with limited labelled examples [24].

- **The metric learning approach:** in few-shot learning, the metric learning strategy focuses on learning a distance metric or similarity measure between samples to evaluate their similarity or dissimilarity [25]. This method seeks to allow for effective comparison of query samples with the few labelled instances that are available (i.e., learning to compare). Several metric-based approaches have been proposed in few-shot learning such as the following:
  - **Siamese networks:** in Siamese networks, embeddings for pairs of samples are learned by training a network with weights that are the same for both instances. The goal of the network is to reduce the distance between samples that are alike and increase the distance between samples that are different. This lets the network learn a space where samples that are similar are close together and samples that are different are farther apart [26].
  - **Matching networks:** use an attention mechanism to weigh the contribution of each labeled example to the classification decision for new examples. The attention mechanism is learned based on a few labeled examples from each class, which enables the model to generalize to new tasks with minimal labeled data [27].
  - **Prototypical networks:** is a popular metric-based few-shot learning technique that uses a distance metric to classify new examples based on their similarity to prototype examples. Prototypes are generated by computing the mean of the embedding vectors of a few labeled examples in each class. The model then assigns the test example to the class with the nearest prototype [28].

Table 2 Prototypical Networks in FSL [29]



- **Relation networks:** learn to assess the class link between pairs of samples by comparing them. They combine these embeddings and run them through several neural network layers to represent the relationship between a query sample and a support set. The output produced is utilized to forecast the query sample's class label [26].

- **The Augmentation approach:** this technique involves creating additional training instances by using various data augmentation approaches to the current labelled data. By enhancing the variety of the training set, this supplemented data helps to alleviate the problem of a few labelled samples. Augmentation methods such as random cropping, flipping, rotation, or adding noise can be used to produce variations of previously labelled data, allowing the model to generalize better to previously unseen examples [9].
- **Meta-learning approach:** often known as learning to learn, seeks to teach models how to adapt fast to new tasks or domains while having insufficient labelled data. Meta-learning methods are often used to train a model on a number of tasks, each having its own information and evaluation sets. With only a few labelled instances, the model learns to extract task-specific information and generalize from prior tasks to new ones [30].

2.2.1 Few-shot Learning-based Text Classification

Recently, a number of studies have emerged that expressly address the challenges associated with few-shot text classification difficulties. [31] Claimed that the choice of an ideal meta-model could differ depending on the specific FSL tasks at hand. To address this, they utilized a multi-metric model, which involved clustering the meta-tasks into distinct clusters with predefined characteristics. [32] Proposed a few-shot text classification model specifically designed for multi-label text classification tasks with a pre-existing label space structure. [33] Proposed a framework for brief text categorization based on Siamese Convolutional Neural Networks (CNNs) and few-shot learning. Siamese CNNs will learndiscriminative text encoding to assist classifiers in distinguishing obscure or informal sentences. The various phrase forms and subject descriptions are treated as 'prototypes' that will be learned using a few-shot learning technique to improve the classifier's generalization. [34] Suggests the Label Semantic Augmented Meta-Learner (LaSAML) architecture to demonstrate how class label information may be used to derive more distinct feature representations of the input text from a pre-trained language model like BERT and can improve performance when the samples are limited. The authors show that this framework may be integrated into the current few-shot text classification system and they perform numerous trials on the LaSAML-upgraded few-shot text classification system which noticeably performs better than its predecessors. On the other hand, [35] Had explored certain limitations for the transformer models that produce a class distribution for a specific prediction task and have a linear layer on top. They proposed a new technique of text classification through the factorization of arbitrary classification tasks into a general binary classification problem, it imbues the transformer model with the idea of the task at hand. The authors demonstrated experiments in few-shot and zero-shot learning that demonstrate how their method significantly outperforms earlier methods on limited training data and can even learn to predict new classes with no training instances at all. [36] Discussed that label names for text categorization tasks include important information. This data has been used by label semantic-aware algorithms to enhance text classification performance during prediction and fine-tuning. However, there hasn't been much research done on the application of label semantics during pre-training. Therefore, in order to enhance the generalization and data efficiency of text classification systems, the study

suggests Label Semantic Aware Pre-training (LSAP). By conducting secondary pre-training on labelled sentences from a range of domains, LSAP combines label semantics into pre-trained generative models. They design a filtering and labelling pipeline to quickly create sentence-label pairs from unlabeled text since domain-broad pre-training necessitates a vast quantity of data. The authors do studies on topic classification (AG News, Yahoo! Answers) and intent (ATIS, Snips, TOPv2). When it comes to few-shot text classification, LSAP surpasses state-of-the-art algorithms in terms of quality while retaining high-resource environments.

3 PRELIMINARIES

Meta-learning: Recently, in the context of NLP, the problem of zero-shot and few-shot learning has been proposed utilizing meta-learning [37], [38]. Meta-learning, often known as learning-to-learn, seeks to learn models how to adapt fast to new tasks or domains while having insufficient labelled data. Meta-learning methods are often used to train a model on a number of tasks, each having its own information and evaluation sets. With only a few labelled instances, the model learns to extract task-specific information and generalize from prior tasks to new ones. This enables the model to quickly adapt to new few-shot tasks and generate accurate predictions [30]. Consider the following scenario, where we are provided with a set of annotated instances belonging to a collection of classes denoted as $\{D\}$, our objective is to construct a model that can gain knowledge from this training dataset. This information will enable us to generate predictions for new classes, which are similar to the original classes but have limited samples available. The newly introduced classes are part of a distinct collection of classes, denoted as $\{D^{test}\}$, which does not overlap with $\{D^{train}\}$ classes. During the process of meta-learning, we replicate this particular scenario in the meta-training phase, enabling our model to acquire the ability to rapidly acquire knowledge from a limited number of annotations. In order to generate a single meta-training episode, we apply episodic learning.

Episodic learning: In the context of FSL the episodic *N*-way, *K*-shot classification is a common process. A numerous influential meta-learning studies (e.g., [39]–[41]) have highlighted the significance of structuring training data into episodes. These episodes consist of learning tasks that include a restricted quantity of "training" instances (known as the support set) and "testing" instances (known as the query set). Each episode aims to replicate the test-time conditions seen in few-shot learning benchmark tasks by subsampling both classes and data points, therefore it enhances the model's ability to generalize well in a new test dataset. During meta-training, we are considering a distribution $\hat{\varepsilon}$ over potential subsets of classes that is as similar as feasible to the one encountered during evaluation ε. While ensuring that the sets of classes seen during training and evaluation are mutually exclusive as suggested by [40]. The episodic batch B_E, denoted as $B_E = \{S, Q\}$, performs the model meta-tasks and it is generated by a two-step process. Firstly, a subset of classes D^{train} is sampled from the set $\hat{\varepsilon}$. Secondly, an input texts are sampled to form the support set S and query set Q from the set of texts that have classes in D^{train}. The support set S is defined as $S = \{(x_1, y_1), ..., (x_n, y_n)\}$ and the query set Q is defined as $Q = \{(x_1, y_1), ..., (x_m, y_m)\}$, where each pair represents a text with its corresponding class and $y_{n,m} = c$ from the subset classes in D^{train}. The model is updated during meta-training by considering the loss incurred across the testing dataset Q. It is important to mention that the model may be trained on a vast number of meta tasks, resulting in a significant challenge against overfitting. For instance, if a dataset has 15 training classes, this results in a total of $\binom{15}{10} = 3{,}003$ potential 10-way tasks. The construction of support and query sets ensures that they include all the classes inside the set D^{train}, and each class is represented by a specified number of samples, referred to as "shots". Hence, episodes are characterized by three variables: the cardinality of the set of classes denoted as $\boldsymbol{n} = |D_c^{train}|$ referred to as "ways", the number of instances per class in the support set represented by $\boldsymbol{k} = |S_c|$ referred to as "shots", and the number of examples per class in the query set denoted as $\boldsymbol{m} = |Q_c|$. During the process of evaluation, the collection of elements $\{n, k, m\}$ serves to establish the framework and parameters of the task. However, during the training phase, they may be seen as a collection of hyperparameters that govern the process of batch generation which needs to be carefully adjusted [42].

4 METHODOLOGY

This research proposes a modified Prototypical Network for the few-shot topic detection problem that uses a transformer BERT-based model (MARBERT) as a feature extractor. Our model consists of two main phases, Phase (1) Data Preprocessing, and Phase (2) Meta-training for topic detection in a few-shot setting as shown in **Error! Reference source not found.**.

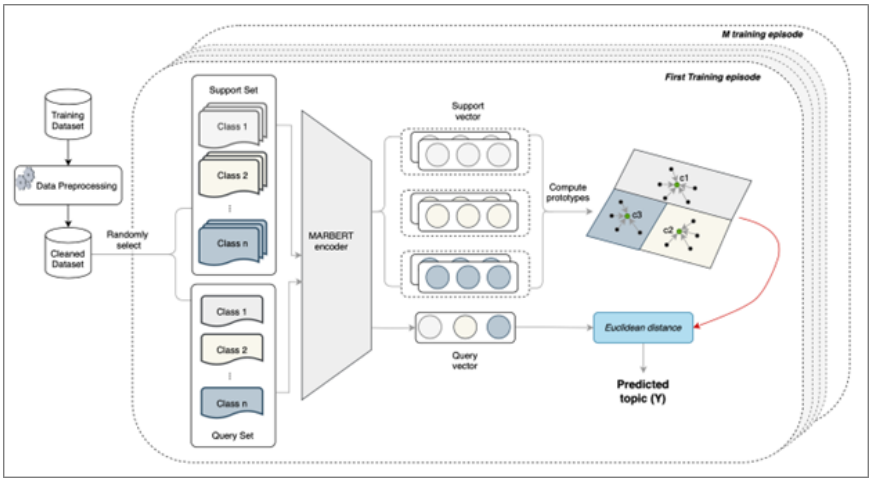


Figure 1 The proposed BP-ASTD Framework

Phase (1) – Data Preprocessing: The preprocessing phase involves cleaning and formatting the text data to ensure it is in an appropriate format for subsequent processing. Phase (2) – Topic detection in a few-shot setting utilizing MARBERT for feature extraction (Meta-training): In this phase the cleaned text resulting from the previous phase is split into a support set and query set in preparation for the random selection of samples (k) through multiple training episodes. For better feature representation and extraction, the transformer BERT-based language model (MARBERT) is used to encode the text data, resulting in a representation that is rich in semantic information [43]. The feature vector for the support set is then averaged to create multiple partitions (prototypes) in the prototypical network. Each prototype represents a class in the utilized dataset and has a centroid (r1, r2, r3…etc.) calculated as an averaged value of each embedding in the support set. During meta-training specified episodes, each encoded vector of the test data, i.e., query data is fed into a similarity function to measure the Euclidean distance between the sample vector and the centroid of each prototype. Where the shorter distance value $-d(X, r_c)$ means that this sample X belongs to the r_c class prototype therefore assign this class as the predicted label for this sample. The algorithm for the proposed approach is given in Algorithm 1.

Algorithm 1: Proposed Model BP-ASTD

Input: (1) Meta-training set $D_c^{train}=\{(x_1, y_1),\ldots,(x_N, y_N)\}$. **(2)** PLM encoder f with parameter θ.

1. for each *Episodic batch* **do:**

2. *Generate a meta-task by randomly select 'n' number of classes associated with its 'k' samples to perform the support set and query set {S, Q}.*

3. for each x_i in S **do:**

4: $Prototype(r_{c'}) \leftarrow ProtoNets\ (f_\theta(x_i))$ **(where c' is a set of all classes within the support set)**

5: **end for**

6: $Loss \leftarrow 0$ **(initialize loss)**

7: **for** each x_j in Q **do:**

8: $Pred_{yc} \leftarrow L2Distance\ (f_\theta(x_j), Prototype\ (r_{c'}))$

9: $loss \leftarrow Compare\ (pred_{yc}, true_{yc})$

10: $loss \leftarrow loss + \left(\frac{-1}{|Q|}\right) . \sum_{(x_j, y_c)\in Q} log\left(\frac{\exp(-d(f_\theta(x_j), r_c))}{\sum_{c'} \exp(-d(f_\theta(x_j), r_{c'}))}\right)$ **(Update loss)**

11: **end for**

12: **end for**

4.1 Pre-processing and Data Cleaning

Preprocessing is the process of reducing the amount of characteristics in the text representation and eliminating noise from the text. It enhances the classifier's performance and improves the reliability and accuracy of the analysis results. Text preparation for the Arabic language encompasses many steps. First, we start with the elimination of symbols like as URLs, hashtags, mentions, retweets, and end-of-line indicators. In addition, The Python NLTK (Natural Language Toolkit) library was used to eliminate Arabic stop words. Special characters like dashes and quotation marks, non-alphanumeric characters, Latin characters, and digits are also replaced with spaces or empty strings. Moreover, diacritical marks or vowel signs were removed to simplify the text. Speech effects, such as the repetition of a single letter many times (e.g., "ممتاااااز"), are often seen in social media postings. These effects serve the primary purpose of emphasizing certain words or phrases. By eliminating additional characters, a unified version of the word may be obtained. Consequently, variations such as "ممتااز" and "ممتاااااااااز" will be transformed into "ممتاز". Tokenization is the subsequent stage in the cleaning process. The process of lexical analysis involves the segmentation of a series of strings into tokens, such as words, phrases, or other linguistic units.

4.2 Feature Extraction and Meta-training

In our framework, we followed the prototypical networks proposed by Snell [29] that apply a CNN as a text encoder. However, in our methodology, we substituted the encoder with the transformer BERT-based model (MARBERT). Based on the experimental results, it has been observed that it outperforms other models when used in conjunction with prototypical networks. In addition to its ability to predict missing words in sentences by considering the words' context from both the left and right sides of a word. Therefore, enables it to understand nuances and ambiguities in Arabic language specifically and extract contextualized embeddings from input texts.

Thus, our model consists of a meta-learner that is composed of two main components: an encoder network denoted as f with learnable parameters θ (MARBERT), and a non-parametric classifier (Prototypical networks) based on distance measurements (i.e., Euclidean distance). Prototypical networks have been seen to possess a higher degree of simplicity and efficiency when compared to more contemporary meta-learning algorithms. As a result, they have garnered significant interest as a viable technique for few-shot and zero-shot learning tasks [80]. During meta-training phase, the embedding function f_θ is provided with random instances from the support set to generate class partitions, also known as prototypes. A prototype *(r)* is calculated for each class *(c)* by taking the average embedding of the samples from the support set that corresponds to that class, computed as follows:

$$r_c = \frac{1}{|S_c|} \sum_{(x_i, y_c) \in S_c} f_\theta\,(x_i)$$

Where the pair (x_i, y_c) represent an instance associated with its label belonging to the support set of a certain class c, and $|S_c|$ refers to the total number of instances per that class. In the context of few-shot classification, the purpose of the support set is to facilitate the learning process, while the query set is used for inference. The objective is to identify the class of the queries based on the labeled supports. Considering the distance function, the nonparametric learner generates a probability distribution across all classes *(c')* for a given query point (x_j) by using a softmax function that operates on the distances between the query point and the prototypes in the embedding space.

$$p_\phi\,(y = c|x_j) = \frac{\exp(-d(f_\theta(x_j)\,, r_c))}{\sum_{c'} \exp(-d(\,f_\theta(x_j), r_{c'}))}$$

Where $-d(f_\phi(x_j), r_c)$ represents the squared Euclidean distance metric, which measures the distance between query samples embeddings and each prototype centroid. Therefore, the closeness of that sample to the center of a class c is indicative of its likelihood of belonging to that class. To proceed in the learning process, we compute the loss after predicting the classes for each query point by minimizing the negative logarithm of the probability of the correct class on each training episode, as follows:

$$J = -\log p_\theta\,(y = c|x_j)$$

$$J = \left(\frac{-1}{|Q|}\right) \cdot \sum_{(x_j, y_j) \in Q} log\left(\frac{\exp(-d(f_\theta(x_j), r_c))}{\sum_{c'} \exp(-d(f_\theta(x_j), r_{c'}))}\right)$$

Where c' is an index that iterates across all classes, therefore, this loss is used to train the encoder network weights $\boldsymbol{\theta}$ using backpropagation.

5 EXPERIMENT

In this study, the objective was to assess the efficacy of Arabic short-text topic detection within the context of few-shot learning. In order to achieve this objective, we conducted experiments using three distinct datasets in Arabic language short-text: SemEval [44], ASND and AITD [45].

5.1 DATASETS

- **SemEval [44]**: Semeval 2017 gold dataset offering a wide variety of Arabic language-related topics. Originally the dataset consisted of 34 distinct topics, covering a wide range of subjects including countries, sports, technology, personalities and others. Some topics were labeled differently although they belong to the same category for example (سوريا, سورية) both refer to Syria with different letters formation, thus we combined tweets of such cases to enable generating a proper dataset for the few-shot learning approach. So, the overall topics in this dataset are reduced to 31. The range of diverse topics allowed us to assess the effectiveness as well as durability of our model across a variety of domains. In our experiment, the dataset was partitioned into three distinct subsets approximately 50% training, 20% validation, and 30% testing.

Table 3 Topics Splits for SemEval Training, Validation and Testing dataset

Training		Validation		Testing	
Topics	**#Tweets**	**Topics**	**#Tweets**	**Topics**	**#Tweets**
Israel	63	Apple	51	Saudi Arabia	36
Iran	63	Android	44	Iraq	25
Erdogan	58	Harry Potter	53	Amazon	35
Terrorism	96	Hillary Clinton	43	Windows10	32
Islam	74	iPhone	46	Real Madrid	28
Obama	57	Justin Bieber	48	Syria	32
Donald Trump	55			Bashar al-Assad	40
Ramadan	64			Barcelona	25
Sissi	69			Pokemon	34
Gucci	56			Aleppo	24
Federer	80				
Messi	69				
Beyoncé	68				
Google	78				
ISIS	93				

- **Arabic Social Media News Dataset (ASND) [45]**: ASND is a comprehensive collection of Arabic social media posts, it includes data from the official YouTube, Facebook, and Twitter accounts of Aljazeera News Channel spanning from February 2017 to September 2019. The dataset consisted of around 6,000 tweets, 2,000 Facebook posts, and 2,000 YouTube video titles. The annotation assignment was conducted via the services of Amazon Mechanical Turk, with the involvement of 12 predetermined categories. During the experiment, three distinct subsets were generated from the dataset. 50%, 17%, and 33% for training, validation, and testing respectively.

Table 4 Topics Splits for ASND Training, Validation, and Testing Dataset

Training		Validation		Testing	
Topics	**#Samples**	**Topics**	**#Samples**	**Topics**	**#Samples**
Art-And-Entertainment	431	Science-and-technology	216	Health	196
Human-Rights-Press Freedom	421	Business-and-economy	202	Education	81
Crime-War-Conflict	1,112			Environment	152
Others	966			Spiritual	96
Politics	4,234				
Sports	239				

- **Arabic Influencer Twitter Dataset (AITD) [45]**: AITD is a dataset compiled from important Arabic Twitter accounts. In this dataset, a field professional identified 60 prominent Arab Twitter influencers, each of whom focuses predominantly on specific content categories. Twitter API was used to retrieve the most recent 3,200 messages from each account, resulting in 10 distinct categories with 115,692 Arabic tweets. In the conducted experiment, 20% of the dataset was dedicated for validation, while 40% was specifically designated for training, and the other 40% was reserved for testing.

Table 5 Topics Splits for AITD Training, Validation, and Testing Dataset

Training		Validation		Testing	
Topics	**#Samples**	**Topics**	**#Samples**	**Topics**	**#Samples**
Spiritual	21,128	Health	9,456	Politics	9,369
Human-Rights-Press Freedom	18,518	Art-and-entertainment	6,247	Education	498
Business-and-economy	10,049			Science-and-technology	4,936
Sports	16,777			Environment	5,010

5.2 IMPLEMENTATION DETAILS

As previously mentioned, we apply a metric-based nonparametric approach, prototypical networks (PN). According to Snell, Swersky, and Zemel (2017), it is recommended to maintain an equal number of shots "*k"* during meta-training and meta-testing phases. Thus, we choose 1,5 and 10 shots in all our experiments for the number of samples in support sets. However, regarding the query set; [46] suggests that Q is typically determined by the user and encompasses a range of 5 to 15 samples per class. Therefore, we retained 5 samples for query sets in all experiments. As per Snell, Swersky, and Zemel (2017), using a greater number of classes "*n"*, or a higher way, during training episodes offers distinct benefits compared to utilizing fewer classes, as it enhances the network's ability to generalize. This is because it compels the model to make more precise judgments in the embedding space, resulting in improved performance. Consequently, in 1-shot, 5-shot, and 10-shot scenarios the training, validation, and testing splits of classes respectively were (15, 6, 10) for SemEval dataset, (6, 2, 4) for ASND, and (4, 2, 4) for AITD. It is worth noting that all these splits contain disjoint classes. The MARBERT encoder was used as a text encoder for both support and query samples, and it was trained using ADAM optimizer [47], with a learning rate of $2x10^{-5}$ and a batch size of 1. To avoid overfitting, every 10 episodes, the learning rate was reduced by a decay weight equal to 0.01 and continue training until the validation loss ceased to show any further improvement. In all our scenarios the training, validation, and testing episodes were randomly sampled as 245,10,100 for SemEval, 300, 20, 250 for ASND, and 500, 30, 150 for AITD. During meta-training phase, the model was meta-validated at intervals of 10 to 30 tasks, and the optimal model was retained based on its performance on the split meta-validation dataset, we used the checkpoint to save the best model. Following meta-training, tasks selected from the meta-testing split were used to meta-test the final model.

6 RESULTS AND DISCUSSION

This section presents a detailed explanation of the results achieved on the benchmark datasets using our proposed approach (BP-ASTD) along with a comparative analysis of several baseline models including MARBERT, ArBERT, BiLSTM, CNN, SVM, NB, KNN, and DT. The selection of these algorithms was based on their effectiveness in the domain of Arabic topic detection over the past ten years [3]. In addition, we compare our results to a meta-learning baseline, the original prototypical networks introduced [29]. The assessment of the model performance in all three datasets was performed on a new test set with novel classes unseen during the training of the model, it involves the identification of class prototypes derived from the support set embeddings of this test set. Subsequently, the class labels of the query point embeddings are predicted via the use of the distance function and a hard classification. The accuracy of the predicted class output is evaluated by comparing it to the real labels of the query points. Since model performance can be sensitive to the number of examples selected for training *k*, we report the average accuracy of 10 experiments. The results of our model performance are shown in Table *6*, and to the best of our knowledge, they currently represent the most advanced achievements on these datasets, utilizing only a few examples and new unseen classes. In terms of 10-shot classification, our model clearly outperforms the original PN introduced in[29].

Table 6 Results of BP-ASTD on SemEval, ASND, and AITD datasets based on novel testing dataset with unseen classes during training of the model

		Dataset		
		SemEval	**ASND**	**AITD**
Model		10-way Acc.	4-way Acc.	4-way Acc.
Our (BP-ASTD)	*1-shot*	67.6%	55.6%	58.6%
	5-shot	80.5%	74.3%	79.3%
	10-shot	86.2%	80.8%	93.4%
PN	*10-shot*	84.3%	76.5%	89.2%

Moreover, we observed that as the number of *k* (shots) increases, there is an average increase in accuracy around 20% for *k* = 1 to *k*=10 across all datasets as illustrated in Figure 2.

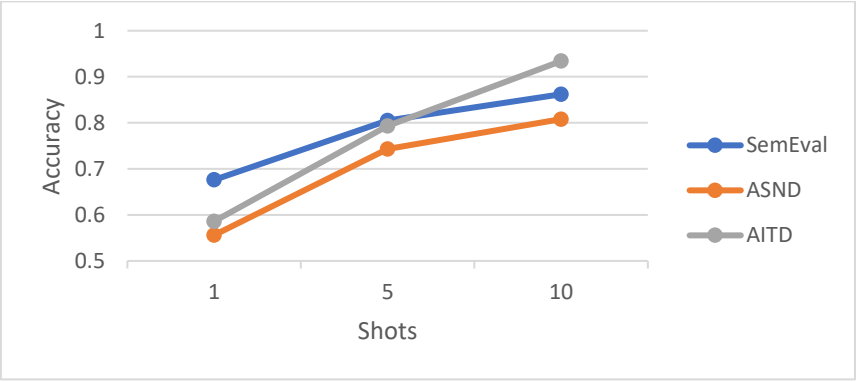


Figure 2 Sensitivity analysis to the number of shots k ∈ {1,5,10} on utilized datasets

In contrast, when comparing to the baseline models the empirical findings shown in Table *7* clearly illustrate that these models perform well due to their traditional ways of utilizing a large quantity of examples for training, and assessing the model considering classes that have been previously observed. However, our model demonstrates superior generalization capabilities when applied to unseen classes and using only a few examples.

Table 7 Accuracy of baseline models on SemEval, ASND, and AITD datasets

Model	Dataset		
	SemEval	**ASND**	**AITD**
MARBERT	**92.1%**	**78.6%**	**93.9%**
BiLSTM	91.9%	68.2%	88.1%
CNN	90.8%	77.3%	92.3%
ArBERT	91.9%	75.2%	91.2%
SVM	87.3%	75.8%	92.3%
NB	76.7%	70.4%	92.5%
KNN	83.9%	66.4%	80.2%
DT	85.7%	62.3%	82.4%

Moreover, according to [24], the prediction accuracy of a model can be influenced by the number of ways *n* in the support set. Specifically, when *n* is increased, the accuracy of the model's predictions tends to decrease. Consequently, we performed a number of experiments on SemEval dataset with $n \in \{3,7,10\}$. Comparing the results of these experiments proves that the selection of the number of *n* strongly affects the model performance as represented in Figure 3, where the best result was achieved in 3-way classification setting. Nevertheless, 7-way and 10-way classification accuracy caused a clear degradation in model performance by approximately 8% and 10% in the 10-shot experiment, as shown in Table *8*.

Table 8 Few-shot Classification Accuracies on SemEval with $n \in \{3,7\}$.

3-way Acc.		**7-way Acc.**	
1-shot	*10-shot*	*1-shot*	*10-shot*
85.5%	96.4%	74.3%	88.8%

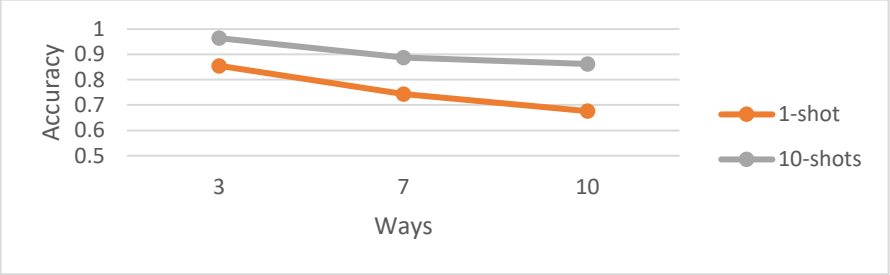


Figure 3 Sensitivity analysis to the number of ways $n \in \{3,7,10\}$ on SemEval dataset

7 CONCLUSION AND FUTURE WORK

This work presents an effective approach for the automated identification of topics in online Arabic short text. The proposed model offers a practical solution for current social media applications seeking to efficiently classify their content, and it is suitable in real-time scenarios where limited data is available for training. The model employs a transformer-based model called MARBERT, which effectively extracts contextualized embeddings for the Arabic language and recognizes complex patterns. In conjunction with the Prototypical meta-learning framework, this model has strong generalization capabilities in the recognition of previously unexplored classes. As demonstrated by the experimental results, the model achieves competitive results compared to several baseline models using only a few examples as 1,5, and 10 to train a classifier that is capable of generalizing well to novel unseen classes. In our prospective research, we aim to explore the application of the model in the domain of Arabic text document classification in addition to including other languages such as English.

REFERENCES

[1] Y. Yang and J. O. Pedersen, "A comparative study on feature selection in text categorization," in Icml, Nashville, TN, USA, 1997, p. 35. Accessed: Nov. 09, 2023. [Online]. Available: http://la.lti.cs.cmu.edu/yiming/Publications/yang-icml97.pdf

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent dirichlet allocation,” J. Mach. Learn. Res., vol. 3, no. Jan, pp. 993–1022, 2003.

[3] A. El Kah and I. Zeroual, “Arabic Topic Identification: A Decade Scoping Review,” in E3S Web of Conferences, EDP Sciences, 2021, p. 01058. Accessed: Nov. 09, 2023. [Online]. Available: https://www.e3s-conferences.org/articles/e3sconf/abs/2021/73/e3sconf_iccsre21_01058/e3sconf_iccsre21_01058.html

[4] M. M. Fouad and M. A. Atyah, “Efficient Topic Detection System for Online Arabic News,” Int. J. Comput. Appl., vol. 975, p. 8887, 2018.

[5] A. Albirini, Modern Arabic sociolinguistics: Diglossia, variation, codeswitching, attitudes and identity. Routledge, 2016. Accessed: Nov. 09, 2023. [Online]. Available: https://books.google.com/books?hl=en&lr=&id=k4SPCwAAQBAJ&oi=fnd&pg=PP1&dq=A.+Albirini,+Modern+Arabic+Sociolinguistics.+2016.&ots=h04PBHOWXr&sig=Zv7MDhR1KgKYFWJBF2KoMBekw00

[6] S. Alzahrani and H. Aljuaid, “Identifying cross-lingual plagiarism using rich semantic features and deep neural networks: A study on Arabic-English plagiarism cases,” J. King Saud Univ.-Comput. Inf. Sci., vol. 34, no. 4, pp. 1110–1123, 2022.

[7] A. Alsanad, “Arabic Topic Detection Using Discriminative Multi nominal Naïve Bayes and Frequency Transforms,” in Proceedings of the 2018 International Conference on Signal Processing and Machine Learning, Shanghai China: ACM, Nov. 2018, pp. 17–21. doi: 10.1145/3297067.3297095.

[8] M. Munir, S. A. Siddiqui, A. Dengel, and S. Ahmed, “DeepAnT: A deep learning approach for unsupervised anomaly detection in time series,” Ieee Access, vol. 7, pp. 1991–2005, 2018.

[9] R. Wei and A. Mahmood, “Optimizing few-shot learning based on variational autoencoders,” Entropy, vol. 23, no. 11, p. 1390, 2021.

[10] R. Koulali, M. El-Haj, and A. Meziane, “Arabic Topic Detection using automatic text summarisation,” in 2013 ACS International Conference on Computer Systems and Applications (AICCSA), 2013, pp. 1–4.

[11] H.-J. Choi and C. H. Park, “Emerging topic detection in twitter stream based on high utility pattern mining,” Expert Syst. Appl., vol. 115, pp. 27–36, 2019.

[12] Y. Chai and W. Li, “Towards deep learning interpretability: A topic modeling approach,” 2019, Accessed: Nov. 09, 2023. [Online]. Available: https://core.ac.uk/download/pdf/301383687.pdf

[13] M. Asgari-Chenaghlu, M.-R. Feizi-Derakhshi, L. farzinvash, M.-A. Balafar, and C. Motamed, “TopicBERT: A Transformer transfer learning based memory-graph approach for multimodal streaming social media topic detection,” Chaos Solitons Fractals, vol. 151, p. 111274, Oct. 2021, doi: 10.1016/j.chaos.2021.111274.

[14] K. E. Daouadi, R. Z. Rebaï, and I. Amous, “Optimizing semantic deep forest for tweet topic classification,” Inf. Syst., vol. 101, p. 101801, 2021.

[15] I. Hmeidi, M. Al-Ayyoub, N. A. Abdulla, A. A. Almodawar, R. Abooraig, and N. A. Mahyoub, “Automatic Arabic text categorization: A comprehensive comparative study,” J. Inf. Sci., vol. 41, no. 1, pp. 114–124, Feb. 2015, doi: 10.1177/0165551514558172.

[16] A. El Mahdaouy, E. Gaussier, and S. O. El Alaoui, “Arabic Text Classification Based on Word and Document Embeddings,” in Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2016, vol. 533, A. E. Hassanien, K. Shaalan, T. Gaber, A. T. Azar, and M. F. Tolba, Eds., in Advances in Intelligent Systems and Computing, vol. 533. , Cham: Springer International Publishing, 2017, pp. 32–41. doi: 10.1007/978-3-319-48308-5_4.

[17] H. M. Abdelaal, A. N. Elmahdy, A. A. Halawa, and H. A. Youness, “Improve the automatic classification accuracy for Arabic tweets using ensemble methods,” J. Electr. Syst. Inf. Technol., vol. 5, no. 3, pp. 363–370, 2018.

[18] A. M. Bdeir and F. Ibrahim, “A Framework for Arabic Tweets Multi-label Classification Using Word Embedding and Neural Networks Algorithms,” in Proceedings of the 2020 2nd International Conference on Big Data Engineering, Shanghai China: ACM, May 2020, pp. 105–112. doi: 10.1145/3404512.3404526.

[19] M. F. Ibrahim, M. A. Alhakeem, and N. A. Fadhil, “Evaluation of Naïve Bayes classification in Arabic short text classification,” Al-Mustansiriyah J Sci, vol. 32, no. 4, pp. 42–50, 2021.

[20] M. M. Fouad and M. A. Atyah, “Efficient Topic Detection System for Online Arabic News,” Int. J. Comput. Appl., vol. 975, p. 8887, 2018.

[21] A. Abuzayed and H. Al-Khalifa, “BERT for Arabic topic modeling: an experimental study on BERTopic technique,” Procedia Comput. Sci., vol. 189, pp. 191–194, 2021.

[22] X. Sun, B. Wang, Z. Wang, H. Li, H. Li, and K. Fu, “Research progress on few-shot learning for remote sensing image interpretation,” IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens., vol. 14, pp. 2387–2402, 2021.

[23] J. Tang, Y. Zhao, L. Feng, and W. Zhao, “Contour-Based Wild Animal Instance Segmentation Using a Few-Shot Detector,” Animals, vol. 12, no. 15, p. 1980, 2022.

[24] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni, “Generalizing from a Few Examples: A Survey on Few-shot Learning,” ACM Comput. Surv., vol. 53, no. 3, pp. 1–34, May 2021, doi: 10.1145/3386252.

[25] V.-N. Tuyet-Doan, T.-D. Do, N.-D. Tran-Thi, Y.-W. Youn, and Y.-H. Kim, "One-shot learning for partial discharge diagnosis using ultra-high-frequency sensor in gas-insulated switchgear," Sensors, vol. 20, no. 19, p. 5562, 2020.

[26] S. Basabain, E. Cambria, K. Alomar, and A. Hussain, "Enhancing Arabic-text feature extraction utilizing label-semantic augmentation in few/zero-shot learning," Expert Syst., vol. 40, no. 8, p. e13329, Sep. 2023, doi: 10.1111/exsy.13329.

[27] M. Khalifa, M. Abdul-Mageed, and K. Shaalan, "Self-Training Pre-Trained Language Models for Zero- and Few-Shot Multi-Dialectal Arabic Sequence Labeling." arXiv, Feb. 02, 2021. Accessed: Nov. 09, 2023. [Online]. Available: http://arxiv.org/abs/2101.04758

[28] M. Hardalov, A. Arora, P. Nakov, and I. Augenstein, "Few-shot cross-lingual stance detection with sentiment-based pre-training," in Proceedings of the AAAI Conference on Artificial Intelligence, 2022, pp. 10729–10737. Accessed: Nov. 09, 2023. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/21318

[29] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," Adv. Neural Inf. Process. Syst., vol. 30, 2017, Accessed: Nov. 09, 2023. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/hash/cb8da6767461f2812ae4290eac7cbc42-Abstract.html

[30] B. Liang et al., "Few-shot Aspect Category Sentiment Analysis via Meta-learning," ACM Trans. Inf. Syst., vol. 41, no. 1, pp. 1–31, Jan. 2023, doi: 10.1145/3529954.

[31] M. Yu et al., "Diverse Few-Shot Text Classification with Multiple Metrics." arXiv, May 19, 2018. Accessed: Nov. 09, 2023. [Online]. Available: http://arxiv.org/abs/1805.07513

[32] A. Rios and R. Kavuluru, "Few-Shot and Zero-Shot Multi-Label Learning for Structured Label Spaces," Proc. Conf. Empir. Methods Nat. Lang. Process. Conf. Empir. Methods Nat. Lang. Process., vol. 2018, pp. 3132–3142, 2018.

[33] L. Yan, Y. Zheng, and J. Cao, "Few-shot learning for short text classification," Multimed. Tools Appl., vol. 77, no. 22, pp. 29799–29810, Nov. 2018, doi: 10.1007/s11042-018-5772-4.

[34] Q. Luo, L. Liu, Y. Lin, and W. Zhang, "Don't miss the labels: Label-semantic augmented meta-learner for few-shot text classification," in Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, 2021, pp. 2773–2782. Accessed: Nov. 09, 2023. [Online]. Available: https://aclanthology.org/2021.findings-acl.245.pdf

[35] K. Halder, A. Akbik, J. Krapac, and R. Vollgraf, "Task-aware representation of sentences for generic text classification," in Proceedings of the 28th International Conference on Computational Linguistics, 2020, pp. 3202–3213. Accessed: Nov. 09, 2023. [Online]. Available: https://aclanthology.org/2020.coling-main.285/?utm_campaign=revue&utm_medium=email&utm_source=Revue%20newsletter

[36] A. Mueller et al., "Label Semantic Aware Pre-training for Few-shot Text Classification." arXiv, May 29, 2022. Accessed: Nov. 09, 2023. [Online]. Available: http://arxiv.org/abs/2204.07128

[37] R. Geng, B. Li, Y. Li, X. Zhu, P. Jian, and J. Sun, "Induction Networks for Few-Shot Text Classification." arXiv, Sep. 29, 2019. Accessed: Nov. 09, 2023. [Online]. Available: http://arxiv.org/abs/1902.10482

[38] X. Han et al., "FewRel: A Large-Scale Supervised Few-Shot Relation Classification Dataset with State-of-the-Art Evaluation." arXiv, Oct. 26, 2018. Accessed: Nov. 09, 2023. [Online]. Available: http://arxiv.org/abs/1810.10147

[39] O. Vinyals, C. Blundell, T. Lillicrap, and D. Wierstra, "Matching networks for one shot learning," Adv. Neural Inf. Process. Syst., vol. 29, 2016, Accessed: Nov. 09, 2023. [Online]. Available: https://proceedings.neurips.cc/paper/2016/hash/90e1357833654983612fb05e3ec9148c-Abstract.html

[40] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in International conference on machine learning, PMLR, 2017, pp. 1126–1135. Accessed: Nov. 09, 2023. [Online]. Available: https://proceedings.mlr.press/v70/finn17a.html

[41] S. Ravi and H. Larochelle, "Optimization as a model for few-shot learning," in International conference on learning representations, 2016. Accessed: Nov. 09, 2023. [Online]. Available: https://openreview.net/forum?id=rJY0-Kcll

[42] S. Laenen and L. Bertinetto, "On episodes, prototypical networks, and few-shot learning," Adv. Neural Inf. Process. Syst., vol. 34, pp. 24581–24592, 2021.

[43] M. Abdul-Mageed, A. Elmadany, and E. M. B. Nagoudi, "ARBERT & MARBERT: Deep Bidirectional Transformers for Arabic." arXiv, Jun. 07, 2021. Accessed: Nov. 09, 2023. [Online]. Available: http://arxiv.org/abs/2101.01785

[44] S. R. and N. F. and P. Nakov, "{SemEval}-2017 Task 4: Sentiment Analysis in {T}witter}," in Proceedings of the 11th International Workshop on Semantic Evaluation, Association for Computational Linguistics, 2017.

[45] S. A. Chowdhury, A. Abdelali, K. Darwish, J. Soon-Gyo, J. Salminen, and B. J. Jansen, "Improving Arabic text categorization using transformer training diversification," in Proceedings of the fifth arabic natural language processing workshop, 2020, pp. 226–236. Accessed: Nov. 09, 2023. [Online]. Available: https://aclanthology.org/2020.wanlp-1.21/

[46] D. Das and C. G. Lee, "A two-stage approach to few-shot learning for image recognition," IEEE Trans. Image Process., vol. 29, pp. 3336–3350, 2019.

[47] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization." arXiv, Jan. 29, 2017. Accessed: Nov. 09, 2023. [Online]. Available: http://arxiv.org/abs/1412.6980