rematchka at NADI 2023 shared task: Parameter Efficient tuning for Dialect Identification and Dialect Machine Translation

Reem Abdel-Salam
Cairo University, Faculty of Engineering, Computer Engineering / Giza, Egypt
reem.abdelsalam13@gmail.com

Abstract

Dialect identification systems play a significant role in various fields and applications as in speech and language technologies, facilitating language education, supporting sociolinguistic research, preserving linguistic diversity, and enhancing text-to-speech systems. In this paper, we provide our findings and results in the NADI 2023 shared task for country-level dialect identification and machine translation (MT) from dialect to MSA. The proposed models achieved an F1-score of 86.18 at the dialect identification task, securing second place in the first subtask. Whereas for the machine translation task, the submitted model achieved a BLEU score of 11.37 securing fourth and third place in the second and third subtasks. The proposed model utilizes parameter-efficient training methods which achieves better performance when compared to conventional fine-tuning during the experimentation phase.

1 Introduction

Dialect identification plays a crucial role in understanding and analyzing linguistic variation within a language. This importance extends to the Arabic language, which encompasses a wide range of dialects spoken across various regions. With the advancements in natural language processing and language models, dialect identification systems have become increasingly valuable in accurately identifying and distinguishing Arabic dialects. By accurately identifying Arabic dialects, language models contribute to fields such as speech recognition, language learning, and even cultural preservation. However, Dialect identification in the Arabic language presents unique challenges due to the extensive linguistic diversity and complexity of Arabic dialects. Language models, while powerful tools for natural language processing, face inherent difficulties when applied to Arabic dialect identification. These challenges arise from dialectal variations, limited training data, and data scarcity for certain dialects. The NADI shared task series (Abdul-Mageed et al., 2020, 2021b, 2022) is a well-known competition that offers datasets and modeling opportunities in order to improve research work developed for dialect identification. In previous versions of the competitions, various teams have participated. (Messaoudi et al., 2022) fine-tuned MARBERT using two different approaches. The first approach uses model embedding along with a CNN classifier. The other approach is to use model embedding with quasi-recurrent neural networks. (Abdel-Salam, 2022) used is an ensemble between fine-tuned BERT-based models and various approaches of parameter efficient tuning including p-tuning and prompt-tuning. (Bayrak and Issifu, 2022) used general pre-training as a first step followed by fine-tuning. AlKhamissi et al. (2021) added an adapter layer on top of the MARBERT model.

This paper presents our work and findings in the NADI 2023 shared task (Abdul-Mageed et al., 2023). The NADI 2023 shared task consists of three subtasks. The first subtask is a country-level dialect identification, while the second and third subtasks are a sentence-level machine translation from four dialects to MSA, given that a key challenge is the hard nature of the problem. We use best practices from recent research on improving model generalization and robustness by using different parameter-efficient techniques (PEFT). Parameter-efficient fine-tuning (PEFT) is an alternative to full model fine-tuning, where a small number of task-specific parameters are updated and the majority of language model parameters are frozen. In this way, only one general language model alongside the modified parameters for each task is saved or transferred. PEFT techniques include Prefix-tuning (Li and Liang, 2021), LoRa (Hu et al., 2021), Prompt-tuning (Lester et al., 2021) and Soft-prompting (Liu et al., 2023). The rest of the paper is structured as follows: section 3 discusses the proposed methods,

Proceedings of the The First Arabic Natural Language Processing Conference (ArabicNLP 2023), pages 652–657

section 4 shows experimental results, and section 5 concludes the paper.

2 Dataset

Subtask 1 of NADI 2023 (Abdul-Mageed et al., 2023) provides training and development sets with 18 country dialects. The training set constitutes 18K instances and the development set 1.8K instances. In the evaluation phase, the test set provided contains 3.6K instances. For subtask 2, the provided dataset was MADAR-parallel-corpus (Bouamor et al., 2018). The training set consisted of 12000 examples, while validation and test sets consisted of 400 and 2,000 examples.

3 Methodology

This section presents the various approaches used while developing the final models. For subtask 1, the final model is a weighted ensemble of PEFT BERT-based models and fine-tuned models. For subtasks 2&3, a single model is used.

3.1 Subtask 1 models

In subtask 1, the goal was to identify 18 different Arabic dialects. In order to tackle this problem, we have experimented with several approaches. Most of the models used were BERT-based models such as MARBERT (Abdul-Mageed et al., 2021a), AraBERT (Antoun et al.), QARiB (Abdelali et al., 2021), AraELECTRA discriminator (Antoun et al., 2021), and CAMeLBERT (Inoue et al., 2021). Multiple methods were used: 1) fine-tuning, 2) prompt-tuning, 3) prefix-tuning, 4) soft-prompting, 5) few-shot with contrastive learning, 6) adapter based fine-tuning, and 7) pre-training followed by fine-tuning. **In prompt-tuning** only prompts are introduced into the input embedding sequence, which is fed to the language model head and output to the linear classification head. One of the difficulties in prompting is the design of the prompt and the model's output. For the prompt we have used [MASK]اللغة هي ("**language is [MASK]**"), and [MASK]تصنيف اللهجات في التغريده (" **the dialect in the tweet is [MASK]**") .'and for the output, we have used country names translated into Arabic, as shown in figure 1. **In Soft-prompting** virtual learnable tokens are inserted into the input embedding sequence along with input text, and then this representation is fed to a classifier head, as shown in figure 2. **In prefix-tuning** virtual learnable tokens are inserted into every layer in the model. **In the few shot** settings we have used 100 samples from each class then we have applied supervised contrastive loss along with cross-entropy loss. **For the pre-training followed by fine-tuning**, we first pre-train BERT-based models on the previous year's dataset, and then we fine-tune the model on the newly provided dataset.

Experimental Set-up For the fine-tuned models the learning rate was set to 3e-5 or 4e-6, a cosine-annealing learning rate scheduler was used, the model's weight decay was set to 1e-2 and the length of the sentence for tokenization was set to 256. During training, batch size was set to 8, and at the end of each epoch, the model was evaluated on dev-set. The best-performing model in terms of F1-micro is saved. In all experiments, the first two layers and the embedding were kept frozen.

Submitted systems For this subtask, three different systems were submitted. The first system is a weighted ensemble of all models listed in table 2. For determining the weights of each, we used an optimization method, where the goal is to find the best set of weights that minimize log-loss between the weighted prediction of all models and the true labels of the dev-set. For the second and the third system, we have chosen the best combination of models that yields a high F1-score in the dev-set, through an exhaustive search, as well as optimization to determine the best set of ensemble weights. The experiment goes as follows: we first generate each possible combination of the developed models. Then for each combination, we apply an optimization scheme to determine the best set of weights for each model based on the F1-score calculated between the weighted prediction and actual labels of the dev-set. Finally, we choose the best combination that yields the best F1 score. The models for the second system were: MARBERT with adapter layer, MARBERT with prefix tuning, CAMeLBERT, and QARiB. The models for the third system were: MARBERT with prompt-tuning, MARBERT with soft prompting, MARBERT with prefix-tuning, and MARBERT with pre-training and then fine-tuning.

3.2 Subtask 2&3 models

In this subtask, the goal is to translate a dialectal sentence into MSA. To tackle this problem we have experimented with several approaches in the development phase (dev-phase). The model used

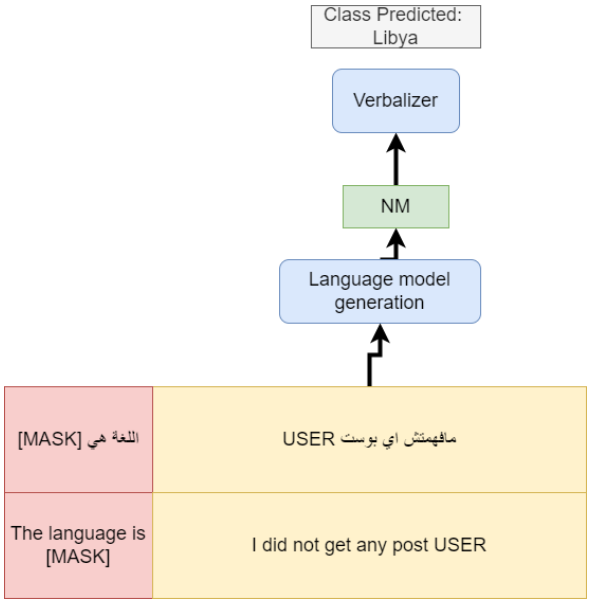

Figure 1: Prompt-tuning architecture.

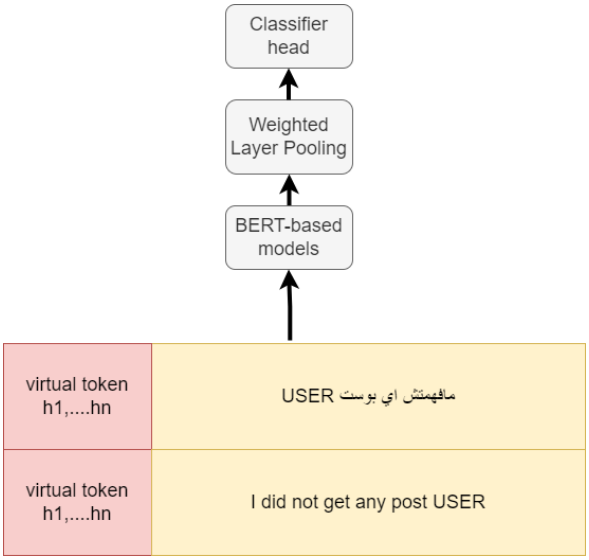

Figure 2: Soft-prompting architecture.

was AraT5v2 (Nagoudi et al., 2022). Several methods have been investigated as 1) conventional-fine-tuning, 2) LoRa, and 3) prompt-tuning. **In LoRa** instead of fine-tuning all the weights that constitute the weight matrix of the pre-trained large language model, two smaller matrices that approximate this larger matrix are fine-tuned. These matrices constitute the LoRA adapter. This fine-tuned adapter is then loaded to the pre-trained model and used for inference. **In prompt-tuning** the following prompt was added before each text to be translated أعد صياغه الجمله للعربيه الفصحى . **(“Rephrase the following to modern standard Arabic")** another prompt investigated was text followed by **source dialect => target dialect, example: CAI => MSA.**.

Experimental Set-up In all of the configurations the encoder and decoder embedding were frozen. The learning rate was set to 6e-6, with a model weight decay of 1e-2. Linear learning rate scheduler was used and the length of the sentence for tokenization was set to 256. Models were fine-tuned for 10 epochs with a batch size of 2. The best-performing model in terms of BLEU score is saved. For LoRa, the following parameters were used: the scaling factor was set to 4, while the rank: was set to 1.

Submitted systems In these subtasks, only one submission was made based on the conventional-fine-tuning method.

4 Results and Discussion

In this section, the performance of the models is reported based on the official metric during dev-phase and test-phase. Moreover, error analysis is conducted to identify weaknesses of the proposed models. For subtask 1 the official metric is the micro average F1-score, while for subtask 2&3 the official metric is the BLEU score.

4.1 Dev-phase results

Table 2 shows results on dev-set for subtask 1. It can be concluded that prompt-based model performed better than fine-tuning methods, prefix-tuning, and soft prompting. The margin difference is around 1%. Table 3 shows submission scores based on the F1-score on the dev-set. All model ensemble underperforms when compared to selective model ensemble. On the other hand, it takes a lot of time to search all possible combinations to select the best one. During experimentation, the model performance decreased while using a combined dataset of the previous year’s dataset and the current year’s dataset, compared to using only this year’s dataset. Our key findings were: PEFT techniques outperform conventional fine-tuning by a magnitude of a maximum of 8% and a minimum of 3%. Prompt-based models were the best-performing models in PEFT, however, they are sensitive to the prompt used. For instance, the results when using the prompt [MASK]اللغة هي (“**language is [MASK]**"), outperform the results from [MASK]تصنيف اللهجات في التغريده (“ **the**

dialect in the tweet is [MASK]") by a magnitude of 1%. Table 1, shows the BLEU score achieved using different techniques for subtask 2&3. LoRa shows significant performance compared to other techniques such as prompting and conventional fine-tuning, with a margin of 3%. This might be due to the fact that the prompt needs more engineering and the hyperparameters re-adjustment. For instance, to our surprise, the second prompt achieved better performance than the first prompt, described in section 3.2. During experimentation, the model showed high sensitivity to the learning rate and weight decay. For instance, we have conducted 3 runs for each experimentation. In the setup, all configurations were kept the same except for the learning rate. The learning rate was set to 1e-6, 3e-6, 6e-6. There were high variation in the results by a magnitude of 2%. For the experiment with a learning rate of 1e-6 the BLEU score was around 8, for a learning rate of 3e-6 the score was around 9, and for a learning rate 6e-6, the score increased to 11.

Model	Technique	BLEU score
AraT5	Conventional	11.136
	LoRA	11.04
	Prompting with prompt Rephrase the following to modern standard Arabic	8.54
	Prompting with prompt source dialect =>target dialect	13.503

Table 1: Models and techniques developed during the experimental phase for subtask 2&3.

4.2 Test-phase results

Table 4 and 5 show the performance of the submitted model in the test-phase for all subtasks. For subtask 1, most models had a near performance with a 0.1 present error, unlike in the dev-set. However, top-performing systems in dev-phase are not the same during the test-phase. For instance, submission-2 and submission-1 interleaved places. Although there is a margin difference of 0.02 in the dev-phase, this changes to 0.001 in the test-phase.

4.3 Error Analysis

Further investigations have been carried out to analyze the potential limitations of the system. As seen in Figure 3, our model performs well when predicting most dialects. However, the model confuses between Kuwait and Bahrain, as well as Syrian and Lebanese dialects. . We believe this is due to the geographic natures between those dialects, as these countries are geographically near each other. Thus it is hard to distinguish between them. For subtask 2&3 one of the major problems was slow convergence of the model in the translation task and fast overfitting.

Model	Technique	F1-Score
MARBERT	Prefix-tuning	0.859
	Adapter	0.755
	Soft-Prompt	0.857
	Prompt-tuning	0.83
	pre-training then fine-tuning	0.828
AraBERT v2	Prompt-tuning	0.857
CAMeLBERT	Prefix-tuning	0.76
QARiB	Fine-tuning	0.77
AraELECTRA	Fine-tuning	0.77

Table 2: Models and techniques developed during the experimental phase for subtask 1.

5 Conclusion

We presented our attempts for the NADI shared task in this article. Our solution is an ensemble of many BERT-based models. These models are created in a variety of ways, including prefix-based models, fine-tuned models, and prompt-based models. The findings reveal that our suggested models perform well in the three subtasks, taking second place in subtask 1 and fourth and third places in subtask 2&3. Future work will concentrate on developing a robust model to improve dialect recognition. Furthermore, to research and identify traits that better distinguish dialects.

References

Reem Abdel-Salam. 2022. Dialect & sentiment identification in nuanced Arabic tweets using an ensemble of prompt-based, fine-tuned, and multitask BERT-based models. In *Proceedings of the The Seventh Arabic Natural Language Processing Workshop (WANLP)*,

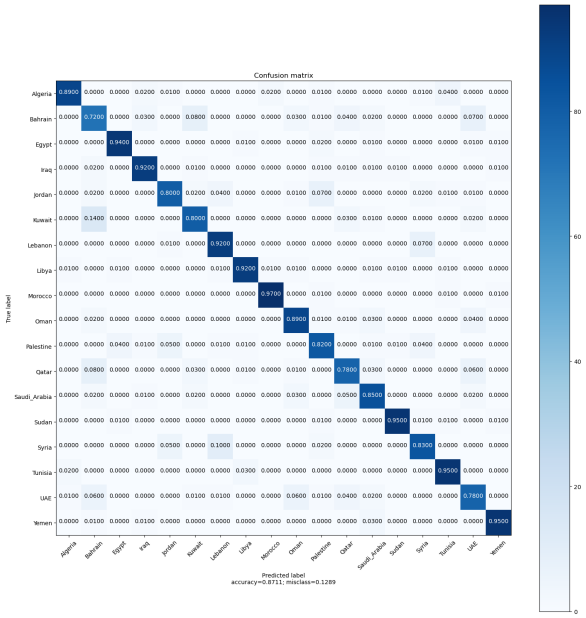

Figure 3: Confusion matrix of the predictions of the submission-3 model in subtask 1 on the dev-set.

Submission	F1-Score
Submission-1	0.859
Submission-2	0.876
Submission-3	0.880

Table 3: Performance of the submitted models on the dev-set in subtask 1. Submission-1 is a weighted ensemble of all models developed, submission-2 is a weighted ensemble of MARBERT with adapter layer, MARBERT with prefix tuning, CAMeLBERT, and QARiB, while submission-3 is a weighted ensemble of MARBERT with prompt-tuning, MARBERT with soft prompting, MARBERT with prefix-tuning, and MARBERT with pre-training then fine-tuning.

Submission Number	F1-Score
Submission-1	0.855
Submission-2	0.853
Submission-3	0.861

Table 4: Performance of the submitted models on the leaderboard in subtask 1. Submission-1 is a weighted ensemble of all models developed, submission-2 is a weighted ensemble of MARBERT with adapter layer, MARBERT with prefix tuning, CAMeLBERT, and QARiB, while submission-3 is a weighted ensemble of MARBERT with prompt-tuning, MARBERT with soft prompting, MARBERT with prefix-tuning, and MARBERT with pre-training then fine-tuning.

Task	BLEU score
Subtask 2	11.37
Subtask 3	11.37

Table 5: Performance of the submitted models on the leaderboard in subtask 2&3

pages 452–457, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Ahmed Abdelali, Sabit Hassan, Hamdy Mubarak, Kareem Darwish, and Younes Samih. 2021. Pre-training bert on arabic tweets: Practical considerations.

Muhammad Abdul-Mageed, AbdelRahim Elmadany, and El Moatez Billah Nagoudi. 2021a. ARBERT & MARBERT: Deep bidirectional transformers for Arabic. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7088–7105, Online. Association for Computational Linguistics.

Muhammad Abdul-Mageed, AbdelRahim Elmadany, Chiyu Zhang, ElMoatez Billah Nagoudi, Houda Bouamor, and Nizar Habash. 2023. NADI 2023: The Fourth Nuanced Arabic Dialect Identification Shared Task. In *Proceedings of The First Arabic Natural Language Processing Conference (ArabicNLP 2023)*.

Muhammad Abdul-Mageed, Chiyu Zhang, Houda Bouamor, and Nizar Habash. 2020. NADI 2020: The first nuanced Arabic dialect identification shared task. In *Proceedings of the Fifth Arabic Natural Language Processing Workshop*, pages 97–110, Barcelona, Spain (Online). Association for Computational Linguistics.

Muhammad Abdul-Mageed, Chiyu Zhang, AbdelRahim Elmadany, Houda Bouamor, and Nizar Habash. 2021b. NADI 2021: The second nuanced Arabic dialect identification shared task. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 244–259, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

Muhammad Abdul-Mageed, Chiyu Zhang, AbdelRahim Elmadany, Houda Bouamor, and Nizar Habash. 2022. NADI 2022: The third nuanced Arabic dialect identification shared task. In *Proceedings of the The Seventh Arabic Natural Language Processing Workshop (WANLP)*, pages 85–97, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Badr AlKhamissi, Mohamed Gabr, Muhammad El-Nokrashy, and Khaled Essam. 2021. Adapting marbert for improved arabic dialect identification: Submission to the nadi 2021 shared task. *arXiv preprint arXiv:2103.01065*.

Wissam Antoun, Fady Baly, and Hazem Hajj. Arabert: Transformer-based model for arabic language understanding. In *LREC 2020 Workshop Language Resources and Evaluation Conference 11–16 May 2020*, page 9.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2021. AraELECTRA: Pre-training text discriminators for Arabic language understanding. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 191–195, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

Giyaseddin Bayrak and Abdul Majeed Issifu. 2022. Domain-adapted BERT-based models for nuanced Arabic dialect identification and tweet sentiment analysis. In *Proceedings of the The Seventh Arabic Natural Language Processing Workshop (WANLP)*, pages 425–430, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Houda Bouamor, Nizar Habash, Mohammad Salameh, Wajdi Zaghouani, Owen Rambow, Dana Abdulrahim, Ossama Obeid, Salam Khalifa, Fadhl Eryani, Alexander Erdmann, and Kemal Oflazer. 2018. The MADAR Arabic dialect corpus and lexicon. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Go Inoue, Bashar Alhafni, Nurpeiis Baimukan, Houda Bouamor, and Nizar Habash. 2021. The interplay of variant, size, and task type in Arabic pre-trained language models. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, Kyiv, Ukraine (Online). Association for Computational Linguistics.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2023. Gpt understands, too. *AI Open*.

Abir Messaoudi, Chayma Fourati, Hatem Haddad, and Moez BenHajhmida. 2022. iCompass working notes for the nuanced Arabic dialect identification shared task. In *Proceedings of the The Seventh Arabic Natural Language Processing Workshop (WANLP)*, pages 415–419, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

El Moatez Billah Nagoudi, AbdelRahim Elmadany, and Muhammad Abdul-Mageed. 2022. AraT5: Text-to-text transformers for Arabic language generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 628–647, Dublin, Ireland. Association for Computational Linguistics.