MOSKA-NLP at AdabEval 2026: Feature-Enriched Ensembling for Arabic Politeness Detection

Nina Andriyanova-Almaamary
College of Computer and Information Sciences, King Saud University
Riyadh, Saudi Arabia
ninamaamary@gmail.com

Abstract

In this paper, we present our system for subtask A of the AdabEval 2026 shared task, which focuses on classifying Arabic text into Polite, Neutral, and Impolite categories. Politeness detection is challenging because it cannot be inferred from lexical meaning alone. This is prominent in Arabic language, where politeness is often conveyed through formulaic expressions, stylistic cues, and dialectal variations. Our approach follows a three-stage strategy. First, we evaluate five Arabic sentence embedding models based on different pretrained encoders to identify a strong representation backbone. Second, we enrich sentence embeddings with explicit lexical, surface-level, and auxiliary signals derived from external models, including dialect, intent, and sarcasm classifiers. Third, we combine predictions from independently trained models, using weighted probability-level ensembling with class-specific decision thresholds to address class imbalance. Experimental results show that feature-enriched representations consistently outperform embedding-only baselines, with additional gains obtained from calibrated ensembling. The proposed system achieves a macro-F1 score of 0.87 and an accuracy of 93% on the official AdabEval 2026 evaluation for subtask A.

Keywords: Arabic NLP, Politeness Detection, Text Classification, Ensemble Learning

1. Introduction

Politeness detection is the task of classifying text according to how polite or impolite it is, typically using labels such as Polite, Neutral, and Impolite. Unlike tasks such as sentiment analysis, politeness cannot be determined from meaning alone. The same message can be polite or impolite depending on how it is phrased, which words are used, and how the person addresses others.

This challenge is prominent in the Arabic language. Politeness is often expressed through greetings, honorifics, religious expressions, and formulaic phrases (Ameri et al., 2023), and can vary across dialects and social context. Polite intent can be conveyed without positive sentiment, while impoliteness can rely on subtle phrasing rather than explicit insults.

The AdabEval 2026 shared task (Alqifari et al., 2026) provides a benchmark for Arabic politeness detection using the ADAB dataset (Al-Khalifa et al., 2026). Systems are evaluated primarily using macro-F1 score, while accuracy is reported as a complementary metric. The task is further complicated by severe class imbalance and indirect expressions of politeness; for example, sarcastic expressions can appear polite on the surface while conveying impoliteness in intent.

In contrast to approaches that rely mainly on end-to-end fine-tuning or LLM prompting, this work focuses on understanding how different combinations of sentence representations, explicit lexical and pragmatic features, and calibrated decision strategies affect performance on Arabic politeness detection. Rather than proposing a new model, our goal is to identify which components contribute most under severe class imbalance and pragmatic ambiguity. Our experiments show that feature enrichment provides the largest gains over embedding-only baselines, while probability-level fusion with class-specific thresholds offers consistent improvements under imbalance. We also find that auxiliary signals such as sarcasm, dialect, and intent are most useful when used as complementary features rather than standalone predictors. On the official test set, the final system achieves a macro-F1 score of 0.87 and an accuracy of 93% ranking second in the shared task. Code is available at: https://github.com/NinaM31/politeness-detection-arabic.

2. Background

2.1. Dataset

We use the ADAB dataset released by AdabEval 2026: Arabic Politeness Detection (Subtask A). The official split contains 4,895 training instances, 693 validation instances, and 1,406 unlabeled test instances. The data includes both Modern Standard Arabic and multiple dialects (Gulf, Egyptian, Levantine, Maghrebi) and spans multiple domains, such as YouTube comments, product reviews, and twitter/X posts. The dataset is highly imbalanced, with Neutral instances accounting for approximately 75% of the data, while Polite and Impolite account for approximately 17% and 9% respectively, as shown in Table 1.

Proceedings of The 7th Workshop on Open-Source Arabic Corpora and Processing Tools (OSACT7) with 5 Shared Tasks, pages 165–173
May 11, 2026.

Split	Polite	Neutral	Impolite
Train	815 (17%)	3664 (75%)	416 (9%)
Valid	115 (17%)	519 (75%)	59 (9%)

Table 1: Label distribution across the official training and validation splits. Percentages are rounded to the nearest integer.

2.2. Task Setup

The task is formulated as a multi-class classification problem. Given a single Arabic sentence as input, the system assigns one of three labels: Polite, Neutral, or Impolite. Systems are evaluated primarily using macro-F1 score, with accuracy reported as a secondary metric.

Politeness is defined by how a statement is expressed rather than by its sentiment polarity; negative content can still be labeled as Polite if phrased respectfully. This makes the task particularly challenging, as it requires capturing pragmatic and stylistic cues beyond surface meaning. To illustrate these distinctions, we provide examples from the ADAB dataset along with English translations:

- **Polite**: Dissatisfaction expressed as a respectful request. Example:
  صعب ولا يعمل بسهوله كل مره افتح التطبيق يتعبني ارجو تطويره
  (*It is difficult and does not work easily every time I open the app it tires me, please improve it*)

- **Neutral**: Dissatisfaction expressed without polite or aggressive cues. Example:
  التحديث الاخير سيء جدا حيث لا يمكنني من الدخول السريع أو الدخول ب الهاتف المصرفي
  (*The latest update is very bad, as it does not allow me to log in quickly or to log in using the banking phone*)

- **Impolite**: Dissatisfaction expressed using insulting cues. Example:
  تطبيق فاشل كان من افضل تطبيقات البنوك في السابق لكن الان لا يرتقي بالخدمه للعملاء
  (*A failed app. It was among the best banking applications in the past, but now it does not rise to the level of customer service*)

2.3. Related Work

Computational politeness research is traditionally grounded in linguistic frameworks such as Brown and Levinson (1987) politeness theory, which models politeness as a pragmatic phenomenon shaped by social context and speaker intent. Prior work has applied these ideas using surface-level cues such as hedging, indirect requests, and politeness markers (Danescu-Niculescu-Mizil et al., 2013).

Research on Arabic politeness detection is limited but has recently started to adopt modern transfer learning and large language models (LLMs). Al-Khalifa et al. (2024) investigate politeness analysis in Arabic social media text, comparing fine-tuned Arabic transformers such as MARBERT (Abdul-Mageed et al., 2021) and CamelBERT (Inoue et al., 2021) with zero-shot and few-shot LLM-based approaches. Their results demonstrate the potential of both paradigms, while highlighting challenges related to pragmatic knowledge, cultural context, emoji usage, and ambiguity in linguistic expression.

In general, Arabic text classification has been dominated by transformer-based models pretrained on a large-scale Arabic corpora. A systematic review by Alammary (2022) shows that MARBERT and AraBERT (Antoun et al., 2020) consistently perform strongly on informal and dialectal Arabic. Recent work on Arabic sentence embeddings, such as General Arabic Text Embedding (Nacar et al., 2025), further demonstrates that sentence transformers trained with contrastive objectives provide robust sentence-level representations for noisy and dialect-rich text, which is particularly important for capturing subtle pragmatic and stylistic distinctions in politeness expressions that are not always reflected in explicit lexical cues.

Our approach follows a three-stage strategy: (i) selecting a strong sentence representation backbone, (ii) enriching representations with explicit lexical and pragmatic features, and (iii) applying probability-level ensembling with class-specific thresholding to address severe class imbalance.

In contrast to prior work that relies primarily on end-to-end fine-tuning or LLM prompting, we combine feature enrichment with auxiliary Arabic-specific signals (dialect, intent, sarcasm) and calibrated thresholding to address pragmatic ambiguity and class imbalance. In particular, auxiliary models are used as feature sources rather than standalone predictors, and class-specific thresholds are applied to improve minority class behavior.

Rather than introducing new modeling components, our contribution lies in identifying which design choices are most effective for Arabic politeness detection and how they interact in practice.

3. System Overview

3.1. Sentence Embedding Backbone

We evaluate five sentence embedding models that differ in their underlying pretrained encoders and representation learning objectives. These include models based on MARBERT and AraBERT, as well as sentence transformers following the matryoshka representations learning paradigm introduced in Nacar et al. (2025). All sentence transformers are initially evaluated in a frozen setting, without task specific fine-tuning. Table 2 reports their validation macro-F1 scores.

Model	Macro-F1
Marbert-all-nli-triplet-Matryoshka(Matryoshka)[1]	0.797
MARBERTv2[2]	0.753
GATE-AraBert-v1[3]	0.714
Arabic-all-nli-triplet-Matryoshka[4]	0.714
Arabic-Triplet-Matryoshka-V2[5]	0.707

Table 2: Frozen validation performance of candidate embedding backbones. Details in Appendix A

Based on frozen macro-F1 scores, `Matryoshka` and `MARBERTv2` are selected as the strongest backbones. While `Matryoshka` achieves higher performance in the frozen setting, `MARBERTv2` yields better results after fine-tuning and is therefore used as the primary backbone. `Matryoshka` is incorporated as a complementary signal within the ensemble. Detailed results of individual fine-tuned models are reported in Table 4.

3.2. Feature Enrichment

Manual lexical indicators. We construct manually curated lexicons of politeness-related markers by combining domain knowledge with inspection of the training data. Candidate expressions include greetings, honorifics, addressee markers, and common insulting terms. To ensure reliability, each keyword is validated on the training set and retained only if it exhibits high class specificity (> 0.75), defined as the proportion of occurrences of a keyword that appear in sentences of its most frequent label, divided by its total occurrences across all classes. The final lexicons contain approximately 30 Polite markers, 40 Impolite markers, 25 Addressee markers, and 25 Honorifics. Table 3 shows representative examples from the manually curated lexicons.

[1] Marbert-all-nli-triplet-Matryoshka
[2] MARBERTv2
[3] GATE-AraBert-v1
[4] Arabic-all-nli-triplet-Matryoshka
[5] Arabic-Triplet-Matryoshka-V2

Category	Sample Markers
Polite terms	شكرا، من فضلك، يعطيك (*thank you, please, give you*)
Addressee markers	يا أخي، يا أختي، حضرتك (*my brother, my sister, sir/ma'am*)
Honorifics	أستاذ، دكتور، سعادة (*professor, doctor, excellency*)
Insult terms	فاشل، كذاب، زبالة (*failure, liar, trash*)

Table 3: Samples from manually curated lexical indicators used for politeness detection.

Automatically derived lexicons. We extract class-specific keywords from the training data by first removing globally frequent terms, appearing in more than 5% of documents, and then selecting words that appear mostly in a single class using frequency ratios. We retain only words that occur frequently (at least 10 times) and are strongly associated with class specificity (> 0.75). This results in dataset-adapted lexicons of size 29 Polite, 351 Neutral, and 5 Impolite.

Surface and auxiliary features. We incorporate lightweight surface and pragmatic cues, including character elongation, punctuation patterns, emoji counts, and diacritic-based signals. Lexical features are computed on both the original text and a normalized variant (alef/ya normalization, diacritic removal, and repetition reduction) to improve robustness to orthographic variation. We additionally include predictions from external pretrained models for dialect[1], intent[2], and sarcasm[3] detection. These signals are used as complementary inputs rather than standalone predictions. As these models are not specifically evaluated on the ADAB dataset, their predictions may introduce noise due to domain differences.

3.3. Primary Classification Arm

After selecting the best-performing backbone and feature configuration, we construct a primary classification arm that serves as the anchor model. This arm uses fine-tuned `MARBERTv2` embeddings combined with the selected features. On top of this representation, we train three linear classifiers: multinomial logistic regression (LR), a Stochastic Gradient Descent (SGD) classifier with log-loss, and Complement Naive Bayes (NB). LR and SGD operate on the full embedding + feature representation,

[1] marbertv2-arabic-written-dialect-classifier
[2] Arabic-bank77-intent-classification
[3] sarcasm-classifier-bert-base-arabic-camelbert

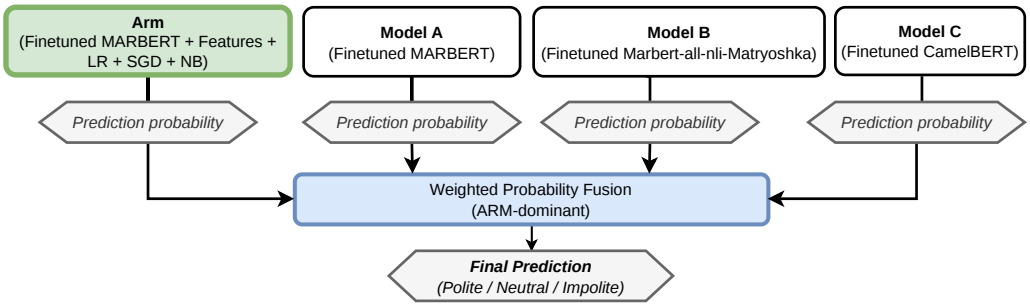

Figure 1: A primary classification arm based on fine-tuned MARBERT embeddings provides the dominant probability estimates, while independently trained auxiliary models contribute complementary signals.

while NB is trained on a sparse feature-only view. Their probability outputs are fused using weighted averaging, with weights 0.44 (LR), 0.46 (SGD), and 0.10 (NB), selected via grid search on the validation set. The SGD classifier is calibrated using sigmoid calibration with 3-fold cross-validation on the training set, while NB serves as a weak complementary signal. The resulting probabilities are then passed to class-specific thresholding to address class imbalance.

3.4. Auxiliary Models and Ensembling

To improve robustness, we incorporate auxiliary models trained with different pretrained backbones. Fine-tuned `MARBERTv2` and `Matryoshka` are included as top-performing models from our backbone experiments, while `CamelBERT`[4] is added based on prior work (Al-Khalifa et al., 2024). These models are not used as standalone predictors but provide complementary probability estimates to the primary arm. Given an input x, each model outputs class probabilities, which are combined via weighted fusion:

$$\mathbf{p}(x) = \sum_{k \in \mathcal{K}} w_k \, \mathbf{p}^{(k)}(x), \quad \sum_{k \in \mathcal{K}} w_k = 1. \qquad (1)$$

The primary arm is assigned a weight of 0.55, while auxiliary models receive smaller weights: 0.10 (MARBERTv2), 0.15 (CamelBERT), and 0.20 (Matryoshka), selected via grid search on the validation set. To address class imbalance, we use class-specific thresholds with Neutral as the default prediction. A prediction is overridden to Polite or Impolite only if the Neutral probability falls below its threshold and the most confident non-neutral class exceeds its corresponding class-specific threshold. The final thresholds are Neutral = 0.32, Polite = 0.40, and Impolite = 0.40, selected using grid search on the validation set. Figure 1 provides an overview of the system.

[4] CamelBERT

4. Experimental Setup

4.1. Hyperparameter Optimization

For frozen sentence embeddings, we train a multinomial logistic regression classifier and tune its hyperparameters through grid search over the regularization strength $C \in \{0.25, 0.5, 1, 2, 4, 8\}$ and solver choice (lbfgs, liblinear). NB is used with a fixed smoothing parameter $\alpha = 0.15$. For fine-tuned models and linear classifiers in the primary arm, hyperparameters are optimized using Optuna, we search over learning rate ($1e^{-5}$–$5e^{-5}$), batch size $\{8, 16\}$, weight decay (0.01–0.1), number of epochs (2–3), and warmup ratio (0.05–0.15), using validation macro-F1 as the objective over 6 trials. For SGD classifiers, we tune regularization strength, penalty type, learning rate schedule, learning rate, maximum iterations, tolerance, and class weighting using Optuna with 30 trials and validation macro-F1 as the objective. The SGD classifier is calibrated using sigmoid calibration with 3-fold cross-validation on the training set. Experiments are implemented using scikit-learn (v1.6.1), Optuna (v2.10.1), and transformers (v4.57.1).

4.2. Feature Selection

Feature selection is performed through ablation experiments on the validation set. We evaluate 45 feature configurations in both frozen and fine-tuned settings. Across both settings, manually curated lexical features, character elongation, and sarcasm signals yield the largest gains, while dialect features provide complementary improvements. The best-performing configuration is used in the final system, with detailed ablation results provided in the Appendix B.

4.3. Ensembling and Thresholding

Final predictions are obtained through weighted probability fusion of the primary classification arm and auxiliary models. The primary arm is assigned the largest weight to preserve stability. Fusion

weights and class-specific thresholds are selected via grid search on the validation set and fixed for test-time evaluation. The final values are reported in Section 3. While this setup is standard in shared task settings, we note that optimizing multiple components on a single validation split may introduce some degree of overfitting to this data.

5. Results

Table 4 summarizes the progressive performance of our system. Starting from frozen sentence embeddings, each component contributes consistent improvements, including fine-tuning, feature enrichment, calibrated classification, and ensembling. The final system achieves a macro-F1 **0.87** and accuracy of **93%** on the official test set, ranking second in the shared task. Ablation experiments confirm that feature enrichment is crucial for Arabic politeness detection. Manually curated lexical features outperform automatically derived lexicons, particularly after normalization. Additionally, character elongation and sarcasm provided robust gains. Aggressively expanding class-specific lexical markers yields diminishing returns. Increasing impolite markers improve validation score up to 0.87 macro-F1, but reduces test performance to 0.86, indicating overfitting and increased cross-class overlap. These findings motivate the conservative feature selection strategy adopted in the final approach.

Stage	Split	Macro-F1	Acc. (%)
Frozen MARBERTv2	Valid	0.753	85
Frozen Matryoshka	Valid	0.797	87
+ Feature enrichment	**Valid**	**0.845**	90
Fine-tuned CamelBERT	Valid	0.822	90
Fine-tuned Matryoshka	Valid	0.834	90
Fine-tuned MARBERTv2	Valid	0.847	90
+ Feature enrichment	Valid	0.853	91
+ Arm (LR + SGD + NB)	Valid	0.859	92
Ensemble (development)	**Valid**	**0.862**	92
Ensemble (official)	**Test**	**0.87**	93

Table 4: Validation results are reported on the development set and official results on the test set. Feature enrichment represents the best configuration, combining manually curated lexical indicators, character elongation, and sarcasm signals. Detailed analysis is provided in Appendix C.

5.1. Error Analysis

We analyze misclassifications on the validation set to identify common error patterns. Most errors involve Impolite and Polite instances being misclassified as Neutral. This is reflected in Figure 2, where 36% of Impolite instances and 14% of Polite instances are predicted as Neutral, while Neutral is correctly classified in 97% of cases. This indicates a strong tendency for the model to default to the Neutral class under ambiguity. A frequent error type involves Impolite instances expressed through evaluative language rather than explicit insults. For example, the sentence خدمة جدا سيئة وتحديث لافائدة منه (*"Very bad service and the update has no benefit"*) is labeled as Impolite but predicted as Neutral, as it lacks clear insulting markers. Lexical ambiguity further contributes to confusion. The adjective سيء (*"bad"*) appears across both Neutral and Impolite contexts, whereas terms such as فاشل (*"failure"*) are consistently associated with impoliteness. These findings highlight the limitations of surface lexical cues and suggest the need for improved modeling of pragmatic context, such as whether a comment is directed at an addressee or expressed as a general statement. Additional error analysis examples are provided in Appendix D.

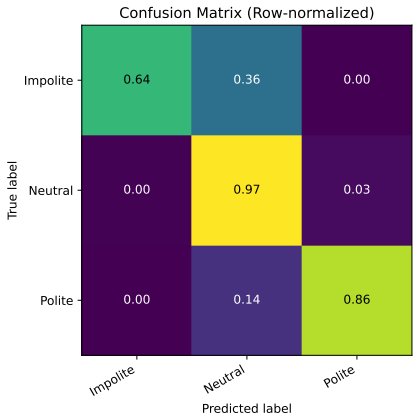

Figure 2: Normalized confusion matrix (validation).

6. Conclusion

We presented a feature-enriched and ensemble-based system for Arabic politeness detection. By combining sentence embeddings, explicit linguistic features, and calibrated decision strategies, the system achieves strong performance on the official test set. Our experiments provide practical insights: feature enrichment yields the largest gains, while auxiliary models are most effective as complementary signals rather than standalone predictors. We note that optimizing multiple components on a single validation split may introduce some overfitting, and reproducibility may depend on access to external pretrained models. The results also highlight the limitations of surface lexical cues, as lexical ambiguity often leads to confusion between classes. Future work includes modeling addressee awareness, improving contextual understanding, and better handling pragmatic ambiguity across dialects.

Acknowledgments

We thank the organizers of the AdabEval 2026 shared task for providing the dataset and timely support. We are also grateful to the anonymous reviewers for their helpful feedback, which improved the quality of this paper. Finally we thank the developers of the open-source pretrained models used in our experiments.

References

Muhammad Abdul-Mageed, AbdelRahim Elmadany, and El Moatez Billah Nagoudi. 2021. 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.

Hend Al-Khalifa, Nadia Ghezaiel, and Maria Bounnit. 2024. Analyzing politeness in Arabic tweets: A preliminary study. In *Proceedings of the 7th International Conference on Natural Language and Speech Processing (ICNLSP 2024)*, pages 352–359, Trento. Association for Computational Linguistics.

Hend Al-Khalifa, Nadia Ghezaiel, Maria Bounnit, Hend Hamed Alhazmi, Noof Abdullah Alfear, Reem Fahad Alqifari, Ameera Masoud Almasoud, and Sharefah Ahmed Al-Ghamdi. 2026. ADAB: Arabic dataset for automated politeness benchmarking – a large-scale resource for arabic politeness analysis. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2026)*, Palma, Mallorca, Spain. European Language Resources Association (ELRA).

Ali Saleh Alammary. 2022. Bert models for arabic text classification: A systematic review. *Applied Sciences*, 12(11).

Reem Alqifari, Hend Al-Khalifa, Nadia Ghezaiel Hammouda, Maria Bounnit, Hend AlHazmi, Ameera Almasoud, Sharefah Al-Ghamdi, and Noof Alfear. 2026. The adabeval 2026 shared task on arabic politeness detection. In *Proceedings of the 7th Workshop on Open-Source Arabic Corpora and Processing Tools (OSACT7) with 5 Shared Tasks*, Palma, Mallorca, Spain. European Language Resources Association (ELRA). Co-located with LREC 2026.

Mohamadali Ameri, Ali Zeighami, and Sayyed Reza Mirahmadi. 2023. A study of the polite method in the arabic language according to brown and levinson's theory of politeness. *Studies on Arabic Language and Literature*, 13(36):61–88.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. AraBERT: Transformer-based model for Arabic language understanding. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 9–15, Marseille, France. European Language Resources Association.

Penelope Brown and Stephen C. Levinson. 1987. *Politeness: Some Universals in Language Usage*. Cambridge University Press.

Cristian Danescu-Niculescu-Mizil, Moritz Sudhof, Dan Jurafsky, Jure Leskovec, and Christopher Potts. 2013. A computational approach to politeness with application to social factors. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 250–259, Sofia, Bulgaria. Association for Computational Linguistics.

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*, pages 92–104, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

Omer Nacar, Anis Koubaa, Serry Sibaee, Yasser Al-Habashi, Adel Ammar, and Wadii Boulila. 2025. Gate: General arabic text embedding for enhanced semantic textual similarity with matryoshka representation learning and hybrid loss training.

Appendix

A. Sentence Embedding Backbones

Table 5 reports the performance of evaluated sentence embedding models in the frozen setting.

Model	Macro-F1	C	Penalty	Solver
Matryoshka	0.797	0.50	l1	liblinear
MARBERTv2	0.753	2.00	l2	liblinear
GATE-AraBert-v1	0.714	0.25	l1	liblinear
A-all-Matryoshka	0.714	2.00	l2	liblinear
A-Matryoshka	0.707	0.25	l2	liblinear

Table 5: Macro-F1 scores for sentence embedding backbones using frozen embeddings and LR.

B. Feature Ablation Results

We conduct extensive ablation experiments to assess the contribution of lexical, surface, pragmatic, and auxiliary features. A total of 45 configurations are evaluated using a frozen encoder and repeated after fine-tuning. This section reports the full results for both settings and summarized in Appendix C.

Feature Groups.

- **Manual lexical (MNL)**: counts of polite phrases, honorifics, insults, and addressee markers.
- **Automatic lexical (Auto)**: count of class-specific keywords extracted from training data.
- **Clean (CLN)**: orthographically normalized variants of lexical features.
- **Surface (ELG)**: character elongation indicators.
- **Pragmatic (PRG)**: emoji count, repeated punctuation, exclamation and question marks, and diacritic statistics.
- **Source Data (SRC)**: text source provided by ADAB (Tweet, Companies, YouTube, Shein).

B.1. Frozen Matryoshka Encoder

This subsection reports ablation experiments using frozen Matryoshka sentence embeddings as the backbone. All feature groups are independently concatenated to the baseline embeddings only.

Main Experiment	Macro-F1	vs. Base (%)
Baseline (Base)	0.797	+0.00
+ SRC	0.796	-0.12
+ Auto	0.803	+0.75
+ SRC + Auto	0.811	+1.83
+ MNL	0.838	+5.20
+ SRC + MNL	**0.841**	**+5.59**

Table 6: Frozen Matryoshka encoder: main experiments obtained by independently concatenating the indicated feature sets to the baseline embedding vector. Improvements are reported relative to the baseline configuration (Base).

Pragmatic Experiment	Macro-F1	vs. Base (%)
Auto CLN + PRG	0.819	+2.80
Auto CLN + ELG + PRG	0.820	+2.91
MNL CLN + ELG + PRG	0.844	+5.89
MNL CLN + PRG	**0.848**	**+6.38**

Table 7: Frozen Matryoshka encoder: pragmatic feature ablation.

Preprocessing Experiment	Macro-F1	Base (%)
ELG	0.801	+0.52
Auto + ELG + SRC	0.812	+1.95
Auto + ELG	0.813	+2.03
Auto CLN	0.815	+2.33
Auto CLN + ELG	0.821	+3.02
MNL + ELG + SRC	0.838	+5.20
MNL + ELG	0.839	+5.32
MNL CLN + ELG	0.845	+6.07
MNL + Auto + ELG	0.846	+6.21
MNL CLN	**0.846**	**+6.23**

Table 8: Frozen Matryoshka encoder: preprocessing ablation results assessing the impact of surface elongation indicators (ELG) and orthographic normalization (CLN) on lexical feature extraction (MNL/Auto), optionally combined with the source metadata feature (SRC).

IDS Experiment	Macro-F1	vs. Base (%)
Auto + I	0.784	-1.55
I	0.785	-1.51
D	0.789	-0.97
S	0.797	+0.00
Auto + D	0.802	+0.68
Auto + S	0.803	+0.75
MNL + I	0.826	+3.63
MNL + Auto + I	0.834	+4.69
MNL + S	0.838	+5.20
MNL + D	0.841	+5.57
MNL + Auto + D	0.846	+6.14
MNL + Auto + S	**0.846**	**+6.21**

Table 9: Frozen Matryoshka encoder: ablation over auxiliary supervision signals for intent (I), dialect (D), and sarcasm (S), evaluated alone and in combination with lexical features (MNL/Auto).

Combined Experiment	Macro-F1	Base (%)
MNL + PRG + IDS	0.824	+3.46
MNL CLN + ELG + IDS	0.828	+3.90
MNL + IDS	0.828	+3.93
MNL + PRG + IDS + SRC	0.830	+4.15
MNL CLN + ELG + PRG + IDS	0.832	+4.42
MNL + PRG + S	0.837	+5.02
MNL + S	0.838	+5.20
MNL + DS	0.840	+5.39
MNL + PRG + DS	0.840	+5.39
MNL CLN + ELG + PRG + S	0.844	+5.89
MNL CLN + ELG + S	0.845	+6.07
MNL CLN + ELG + DS	0.845	+6.07
MNL CLN + ELG + PRG + DS	**0.852**	**+6.90**

Table 10: Frozen Matryoshka encoder: comprehensive feature combinations spanning lexical (MNL/Auto), normalization (CLN), surface elongation (ELG), pragmatic cues (PRG), auxiliary signals (IDS), and source metadata (SRC). The final row highlights the best frozen configuration (Best-M).

B.2. Fine-tuned MARBERT Encoder

This subsection reports the same ablation experiments after fine-tuning MARBERT on the task data. Results are reported relative to the best frozen Matryoshka configuration (Best-M) to facilitate direct comparison.

Main Experiment	Macro-F1	vs. Best-M (%)
Baseline (MARBERT)	0.845	-0.77
+ Auto	0.845	-0.77
+ SRC	0.844	-0.95
+ SRC + Auto	0.847	-0.59
+ MNL	0.849	-0.32
+ SRC + MNL	**0.850**	**-0.14**

Table 11: Fine-tuned MARBERT encoder: main experiments after encoder fine-tuning, where the same feature sets are independently concatenated to the fine-tuned baseline representation.

Pragmatic Experiment	Macro-F1	vs. Best-M (%)
MNL CLN + PRG	0.840	-1.38
Auto CLN + PRG	0.845	-0.77
MNL CLN + ELG + PRG	0.848	-0.49
Auto CLN + ELG + PRG	**0.853**	**+0.12**

Table 12: Fine-tuned MARBERT encoder: pragmatic ablation results measuring the contribution of PRG cues (emoji and punctuation statistics) when combined with normalized/elongation-aware lexical features.

Preprocessing Experiment	Macro-F1	Best-M (%)
Auto CLN	0.845	-0.77
ELG	0.849	-0.33
Auto + ELG	0.848	-0.43
Auto + ELG + SRC	0.849	-0.37
MNL CLN	0.849	-0.32
MNL + Auto + ELG	0.851	-0.10
Auto CLN + ELG	0.853	+0.12
MNL CLN + ELG	0.853	+0.12
MNL + ELG	0.853	+0.12
MNL + ELG + SRC	**0.853**	**+0.12**

Table 13: Fine-tuned MARBERT encoder: preprocessing ablation results evaluating normalization (CLN) and elongation indicators (ELG) under fine-tuning, optionally combined with lexical and source metadata features.

IDS Experiment	Macro-F1	vs. Best-M (%)
Auto + I	0.849	-0.32
I	0.846	-0.73
D	0.847	-0.55
S	0.845	-0.77
Auto + D	0.845	-0.77
Auto + S	0.845	-0.77
MNL + I	0.847	-0.59
MNL + Auto + I	0.847	-0.55
MNL + S	0.849	-0.32
MNL + D	0.849	-0.32
MNL + Auto + S	0.851	-0.10
MNL + Auto + D	**0.853**	**+0.12**

Table 14: Fine-tuned MARBERT encoder: ablation over auxiliary signals for intent (I), dialect (D), and sarcasm (S), evaluated alone and in combination with lexical features.

Combined Experiment	Macro-F1	Best-M (%)
MNL + PRG + DS	0.843	-1.05
MNL + PRG + IDS	0.843	-0.98
MNL CLN + ELG + PRG + IDS	0.844	-0.94
MNL + PRG + S	0.845	-0.77
MNL + PRG + IDS + SRC	0.846	-0.73
MNL CLN + ELG + IDS	0.847	-0.51
MNL CLN + ELG + PRG + DS	0.848	-0.49
MNL CLN + ELG + PRG + S	0.848	-0.49
MNL + IDS	0.849	-0.37
MNL + S	0.849	-0.32
MNL + DS	0.850	-0.14
MNL CLN + ELG + DS	**0.853**	**+0.12**
MNL CLN + ELG + S	**0.853**	**+0.12**

Table 15: Fine-tuned MARBERT encoder: comprehensive feature combinations spanning lexical, preprocessing (CLN/ELG), pragmatic (PRG), auxiliary (IDS/DS), and source metadata (SRC).

C. Experiment Analysis

Across frozen Matryoshka experiments, the lowest-performing configurations rely on auxiliary signals in isolation (e.g., Auto + I: 0.784), while the best performance is achieved by combining lexical, normalization, elongation, pragmatic, and auxiliary features (MNL CLN + ELG + PRG + DS: 0.852), demonstrating that lexical features drive performance, with preprocessing and auxiliary signals providing gains only when integrated as complementary signals. In contrast, fine-tuned MARBERT experiments exhibit smaller performance differences, with the lowest configurations remaining close to the baseline (0.845), while the best combinations (MNL CLN + ELG + DS: 0.853) yield only modest gains. Notably, using the same feature combination (MNL CLN + ELG + DS), frozen Matryoshka achieves 0.845 compared to 0.853 with fine-tuned MARBERT, indicating that fine-tuning provides a consistent advantage while narrowing the gap between embedding backbones.

D. Additional Error Analysis

This section provides a qualitative analysis of representative misclassification cases observed on the validation set. We focus on frequent error types and illustrate them using full Arabic examples, with En-

glish translations, to highlight pragmatic and lexical challenges in Arabic politeness detection.

Impolite predicted as Neutral. The most common error pattern corresponds to Impolite instances misclassified as Neutral. These cases typically express strong dissatisfaction or criticism without explicit insults or abusive language. Negative sentiment is conveyed through evaluative adjectives rather than direct offense, leading the system to favor a Neutral prediction (true label: Impolite, predicted: Neutral):

خدمه سيئة حاولت انشط الحساب الدولي واكلوني رصيد ع الفاضي وقالوا كلمة السر غلط
"Bad service. I tried to activate the international account, they wasted my balance for nothing and said the password was wrong."

خدمة جدا سيئة وتحديث لافائدة منه
"Very bad service and the update has no benefit."

Although these sentences express clear dissatisfaction, they lack explicit insults or addressee markers, causing the model to interpret them as neutral complaints rather than impolite attacks.

Polite predicted as Neutral. Polite instances misclassified as Neutral often rely on implicit politeness strategies rather than explicit polite markers. These include praise, affective language, emojis, or indirect requests. In such cases, politeness is conveyed through tone and social context rather than formal expressions (true label: Polite, predicted: Neutral):

خدمه ممتازه وافتخر بأني من عملاءه
"Excellent service, and I am proud to be one of its customers."

برنامج جميل جدآ وسهل استخداااام
"A very beautiful program and very easy to use."

الدكتور هذا خبير في التغذية
"This doctor is an expert in nutrition."

Despite expressing positive affect or respect, the absence of explicit politeness markers causes the model to default to the Neutral class.

Neutral predicted as Polite. A smaller but notable error category involves Neutral instances predicted as Polite. These cases contain positive adjectives, encouragement, or softening expressions that resemble politeness markers, yet are annotated as Neutral in the dataset (true label: Neutral, predicted: Polite):

اتمنى الحلقة تترجم
"I hope this episode gets translated."

تكفون ابو مشاعل خلوه معكم الموسم كامل معطي جو كبير للبودكاست
"Please, Abu Mishaal, keep him with you for the whole season he adds great energy to the podcast."

While such expressions may appear as polite request, they primarily function as opinions or positive feedback rather than explicit politeness acts. This suggests the presence of borderline or ambiguous annotation cases where pragmatic interpretation is subjective.

Lexical ambiguity in negative evaluative terms. Lexical ambiguity contributes substantially to misclassification. Some evaluative terms are strongly associated with impoliteness, while others appear across multiple classes. For example, the term فاشل ("failure") is consistently used as an impolite marker and is correctly classified in all observed cases. In contrast, the adjective سيء ("bad") frequently appears in neutral product or service complaints, leading to confusion between Neutral and Impolite(true label: Impolite, predicted: Neutral):

سيء جدا كل مره لازم ادخل رقم سري للهاتف المصرفي وفي كل مره يقولي خطاء
"Very bad. Every time I have to enter the banking phone PIN, and every time it tells me it's wrong."

This suggests that not all negative expressions carry the same pragmatic meaning, and that distinguishing between evaluative complaints and targeted impoliteness requires modeling contextual and pragmatic cues beyond surface-level sentiment.