KSAA-2026 Shared Task on Arabic Speech Dictation with Automatic Diacritization

Asma Al Wazrah[1], Waad Alshammari[1], Rawan Almatham[1], Raghad Al-Rasheed[1], Afrah Altamimi[1], Rufael Marew[3], Sawsan Alqahtani[2], Hanan Aldarmaki[3], Abdullah Alharbi[1], Abdulrahman Alshehri[1], Mohamed Assar[1], Amal Almazrua[1], Abdulrahman AlOsaimy[1]

[1]King Salman Global Academy for Arabic Language (KSAA), [2]Princess Nourah bint Abdulrahman University (PNU), [3]Mohamed bin Zayed University of Artificial Intelligence (MBZUAI)
[1,2]Riyadh, Saudi Arabia, [3]Abu Dhabi, United Arab Emirates
[1]{aalwazrah, walshammari, ralmatham, a.altamimi, ralrasheed, aalshehri, masar, aalmazrua, aialharbi, aalosaimy}@ksaa.gov.sa, [2]saalqhtani@pnu.edu.sa, [3]{hanan.aldarmaki, rufael.marew}@mbzuai.ac.ae

Abstract

This paper presents the KSAA-2026 Shared Task on Arabic Speech Dictation with Automatic Diacritization, addressing a persistent challenge in Arabic NLP. The task focuses on transforming speech transcripts into fully diacritized Arabic text by leveraging both the speech signal and its undiacritized transcript. Unlike conventional ASR tasks that focus on transcription, this task integrates acoustic and textual information to improve diacritization accuracy. The shared task consists of two subtasks: (1) Data Contribution, where participants recorded and reviewed speech data through the VoiceWall platform, resulting in 2,160 recordings, and (2) Diacritization, where 5 teams developed systems that generate fully diacritized text from speech and undiacritized transcripts. The dataset includes approximately 5 hours of Modern Standard Arabic (MSA) and multi-dialectal speech with fully diacritized references. Experimental results show that several participant systems outperform the provided baselines, and that incorporating speech information and fine-tuning improves performance compared to text-only approaches. KSAA-2026 shared task establishes a benchmark for multimodal Arabic diacritization and supports the development of robust systems for applications in education, accessibility, and speech-driven text generation.

Keywords: Arabic Diacritization, Speech Dictation, Automatic Speech Recognition

1. Introduction

Arabic poses persistent challenges for NLP and speech processing due to the absence of diacritics in standard text and the wide dialectal diversity across the Arab world. The omission of diacritics introduces significant lexical and syntactic ambiguity, while dialectal variation adds phonological and lexical complexity that further complicates speech transcription.

Text-only diacritization approaches often struggle when applied to speech transcripts, as they fail to capture prosodic and phonetic cues present in the audio signal. At the same time, automatic speech recognition (ASR) systems rarely produce high-quality diacritized outputs (Shatnawi et al., 2024).

To address these challenges, KSAA-2026 shared task introduces a benchmark that integrates both speech signals and undiacritized transcripts as input, encouraging the development of multimodal approaches capable of generating fully diacritized text.

Recent studies further highlight these limitations. Shatnawi et al., (2024) show that text-only models suffer from domain and style mismatches when applied to ASR outputs, while speech-informed approaches lead to significant reductions in error rates. Similarly, Elgamal et al., (2024) emphasize the importance of diacritics in diverse real-world settings, demonstrating that even partial diacritization provides useful signals for improving model performance. The complexity of the task is further illustrated in specialized domains; Alyafeai et al., (2023) report persistent challenges in Arabic poetry, where even state-of-the-art systems exhibit high diacritic error rates.

Motivated by these findings, this competition introduces a community-driven benchmark supported by continuous speech contributions, broader dialectal and speaker coverage, and multimodal modeling approaches. The expected impact spans multiple applications, including educational tools for Arabic learners, speech-driven writing assistants, and improved performance in downstream NLP tasks such as text-to-speech, machine translation, and parsing.

2. Dataset

To encourage community engagement and ensure sustainability, this shared task also includes a data contribution component, which is described in Section 3.1.

The core dataset for the competition consists of approximately 5 hours of Arabic speech audio collected from male and female speakers. The recordings represent a diverse range of nationalities, including Saudi, Egyptian, Kuwaiti, Bahraini, Sudanese, Qatari, Algerian, Syrian, and Palestinian speakers. This diversity ensures broad dialectal coverage alongside Modern Standard Arabic (MSA), making the dataset representative of real-world linguistic variation.

Proceedings of The 7th Workshop on Open-Source Arabic Corpora and Processing Tools (OSACT7) with 5 Shared Tasks, pages 220–224

The dataset spans multiple domains such as politics, economy, sports, religion, and news. Each utterance is relatively short, not exceeding nine seconds, to facilitate alignment and model training.

Speech recordings are paired with corresponding transcripts and fully diacritized references. Basic preprocessing steps, such as normalization and alignment, were applied to ensure data usability.

The dataset is divided into training, development, and test sets as shown in Table 1.

Data split	No. of audio files	Duration
Train	2,327	~4.5 hours
Dev	260	~30 minutes
Test	329	~40 minutes

Table 1: Example of input and output structure.

The training and development sets were released to participants, while the test set was kept hidden to ensure a fair evaluation.

3. Task Description

The shared task is organized into two complementary subtasks designed to address both data collection and model development. The first subtask focuses on expanding the dataset through community contributions, while the second subtask evaluates system performance on automatic diacritization of speech dictation.

3.1 Task-1: Data Contribution

This subtask aims to enrich the shared dataset through open community participation. Anyone can join this track, and participants are invited to record Arabic speech via a dedicated web-based platform (VoiceWall)[1] using diacritized text prompts covering both Modern Standard Arabic (MSA) and a range of dialects.

Participants contribute in two complementary roles:

- **Record (Speak & Contribute):** Participants record speech samples by reading the provided diacritized prompts.
- **Listen and Review:** Participants listen to and review recordings submitted by others to support basic validation and filtering.

The validation process relies on two mechanisms: automated platform checks (e.g., verifying recording completeness, duration, and minimum audio quality thresholds) and cross-team review, whereby each team reviewed and rated the recordings submitted by another team. A recording is considered valid if it passes the automated checks and receives approval from the reviewing team. This structured peer validation ensures a consistent and reproducible quality control process while maintaining broad participation.

3.2 Task-2: Automatic Diacritization of Speech Dictation

Participants in this task are required to build systems that generate fully diacritized Arabic text from speech dictation. Each input consists of a speech signal along with its corresponding undiacritized transcript, and the system output is expected to be a fully diacritized version of the transcript.

The task requires predicting full Arabic diacritics at the character level, including fatḥa, ḍamma, kasra, sukūn, shaddah, and tanwīn marks for each character in the undiacritized text.

Two input settings are considered:

- **Text-only diacritization:** systems receive only the undiacritized transcript and generate the diacritized output.
- **Speech + text diacritization (main track):** systems leverage both the audio signal and the transcript to produce fully diacritized text.

A sample of the input–output data structure (see Table 2) is provided to illustrate the task format. This subtask establishes a benchmark for multimodal diacritization and supports the development of robust tools for education, accessibility, Quranic recitation, poetry, and dialectal speech processing.

Input	Speech Clip	
	Undiacritized transcript	أريد أن أشرب كوبًا من الشاي
Output	Diacritized transcript	أُرِيدُ أَنْ أَشْرَبَ كُوبًا مِنَ الشَّاي

Table 2: Example of input and output structure.

4. Evaluation

The evaluation framework is designed to assess both the data contribution efforts and the performance of diacritization systems. Each subtask is evaluated based on its specific objectives and criteria.

4.1 Task-1: Data Contribution Evaluation

The data contribution is assessed based on participant engagement and the amount of collected speech data. Each participant is

[1] https://falak.ksaa.gov.sa/voicewall/ar/s/falak

assigned 360 text prompts to record using the VoiceWall platform, covering Modern Standard Arabic (MSA) and various dialects.

To support data usability, the platform provides basic validation mechanisms, including listening and review functionalities that allow participants to check recordings and filter out unusable samples. The process is designed to encourage broad participation while maintaining an acceptable level of data quality.

Each submitted recording is associated with both the participant and the contributing team to ensure attribution and transparency. The collected data is aggregated and made available to participants for benchmarking purposes.

4.2 Task-2: Automatic Diacritization of Speech Dictation Evaluation

The submitted systems are evaluated based on their ability to generate fully diacritized Arabic text from speech and undiacritized transcripts.

Evaluation is conducted using the diac evaluation framework (Fadel style)[2], which computes Diacritic Error Rate (DER), Word Error Rate (WER), and Sentence Error Rate (SER), where lower values indicate better performance.

The evaluation is performed under two complementary dimensions. The first dimension considers whether case endings are included or excluded, distinguishing between full syntactic diacritization and core lexical diacritization. The second dimension considers whether characters with no diacritics are included or excluded from evaluation.

5. Systems

5.1 Baseline

We provide three baseline systems corresponding to the task setup. These baselines are intended as reference implementations to illustrate the task and are not optimized for performance.

- **Text-only baseline:** A transformer-based diacritization model trained only on the Tashkeela corpus and operating solely on undiacritized text.
- **Speech + text baseline:** A transformer-based model trained on Tashkeela and CartTTS, replicating the setup of Shatnawi et al., (2024), and using ASR outputs together with undiacritized text, without explicitly incorporating acoustic features.
- **Fine-tuned baseline:** A transformer-based model initialized from the previous setup and further fine-tuned on the training portion of the dataset in the challenge to improve diacritization performance.

5.2 Participating Systems

A total of 29 teams registered for Task-2, of which 5 teams successfully submitted valid results. The participating teams adopted a variety of approaches for the diacritization task. Below is a brief summary of the submitted systems.

Fine-Tashkeel team presents a character-level sequence-to-sequence approach that converts undiacritized text directly into fully diacritized output. The authors do not rely on task-specific fine-tuning for their final submission; instead, they evaluate 18 different model configurations, including text-only, ASR-based, and fine-tuned systems, and select the strongest one. Their analysis shows that text-only Seq2Seq models perform better than several multimodal systems that depend on off-the-shelf ASR. In addition to reporting leaderboard performance, the paper includes detailed error analysis across dialects and shows that dialectal variation and case endings remain the main sources of difficulty.

Eraserhead team presents a speech-aware diacritization pipeline built on a pretrained Transformer model with two inputs: text and ASR output. The main contribution is ASR consistency filtering, where noisy ASR transcripts are not removed but replaced with fallback text when they are judged unreliable, which makes training more stable. The authors train multiple runs with different filtering thresholds, combine them using confidence-based ensemble decoding, and then apply speaker-adaptive post-processing, which adjusts word-final diacritics based on each speaker's estimated style: speakers are classified as SUKUN, VOWEL, or MIXED based on development-set observations, and predictions are adjusted accordingly to reflect speaker-specific word-final diacritic behavior. They also add constrained decoding to ensure that the final output remains aligned with the original input words.

Abjad AI team proposed a multimodal system that extends a character-level Transformer by injecting speech representations through a lightweight fusion method called grouped speech conditioning, which is a mechanism that compresses the speech encoder's frame-level outputs into a small fixed number of pooled tokens (G tokens), projects them into the text representation space, and concatenates them with the text input before feeding the combined sequence into the diacritization model. Instead of using expensive fusion mechanisms such as cross-attention, this approach controls how much speech information is injected while keeping computational costs low. The authors compare different speech backbones, including several Whisper variants and Squeezeformer, and study the effect of varying the number of grouped

[2] https://github.com/rufaelfekadu/Diac

speech tokens (G ∈ {3, 5, 10, 15}). They also use a two-stage training schedule, first freezing the text model and then fine-tuning the whole system, and show that compact speech representations, especially Whisper-small with G=5, are the most effective in this setup.

Thaka team presents the winning system for Task-2, which builds on a multimodal CATT-Whisper architecture combining text and speech representations. Instead of modifying the model architecture, the authors focus on improving the training process using regularization techniques, including R-Drop, Focal Loss, and hyperparameter optimization. The speech encoder (Whisper) is kept frozen, while the text model is fine-tuned on the task data. At inference time, the system applies Monte Carlo Dropout with multiple stochastic forward passes and averages the predictions to improve robustness. The results show that training strategies and regularization play a critical role in improving diacritization performance in low-resource settings, achieving the best performance among all participating systems.

TantaArabNLP team presents a system based on the CATT-Whisper multimodal architecture, combining text and speech representations. The authors explore both text-only and multimodal models, with the best results achieved through fine-tuning the multimodal model on the official dataset. The approach focuses on selective fine-tuning and introduces a post-processing step to preserve punctuation, numbers, and non-Arabic tokens, addressing common errors in model outputs. Experimental results show clear improvements over baselines, achieving a WER of 24.39% and securing second place in the competition.

These diverse approaches highlight different strategies for integrating textual and acoustic information, providing a solid basis for comparative evaluation.

6. Results

The results of the shared task are presented for both subtasks, covering data contribution and automatic diacritization. Each subtask is analyzed based on its specific objectives and evaluation criteria.

6.1 Task-1: Data Contribution Results

A total of 20 teams registered for Task-1, with 6 teams completing the data contribution process and actively submitting recordings through the platform. In total, 2,160 recordings were collected and reviewed. These contributions support the long-term goal of expanding Arabic speech resources. These contributions will be incorporated into future dataset expansions.

6.2 Task-2: Automatic Diacritization Results

The performance of all systems, including participant submissions and organizer baselines, is summarized in Table 3 under four evaluation settings. The primary metric (WER with case endings and including no-diacritic characters) is selected as it provides the most comprehensive and challenging evaluation, requiring accurate prediction of both diacritized and non-diacritized characters, as well as syntactic case endings; it therefore serves as the basis for overall system ranking.

Team	Including no diacritics						Excluding no diacritics					
	With case ending			Without case ending			With case ending			Without case ending		
	*DER	*WER	*SER	*DER	*WER	*SER	*DER	*WER	*SER	*DER	*WER	*SER
Thaka	**6.87**	**23.26**	**66.16**	**5.97**	**17.05**	**53.96**	**6.64**	**18.64**	**64.33**	**4.80**	**10.48**	50.91
TantaArabNLP	7.04	24.39	71.65	6.17	17.96	55.18	6.92	18.97	68.90	5.15	11.06	**50.61**
Abjad AI	7.51	25.34	73.48	6.60	18.66	60.06	7.51	19.95	70.73	5.68	11.85	55.49
Eraserhead	8.23	30.37	80.79	6.76	20.84	64.02	8.46	24.33	74.39	5.87	13.43	55.18
****Fine-tune text + ASR**	9.91	31.84	82.93	7.89	20.99	67.07	8.52	24.73	78.66	4.82	10.89	50.61
Fine-Tashkeel	10.56	34.47	79.88	7.46	20.81	68.29	11.56	29.45	76.83	6.86	14.32	64.33
****text + ASR**	13.50	40.24	82.32	10.58	27.95	71.95	14.26	33.03	75.61	9.96	19.71	60.37
****text only**	17.66	49.85	91.77	13.23	32.24	82.62	20.08	46.20	91.77	13.93	27.07	81.71

*Lower is better

** Baseline

Table 3: Test set results for participant systems and organizer baselines under four evaluation settings.

Among the baselines, the fine-tuned model achieves the best performance, confirming the effectiveness of adapting models to the task-specific training data. However, four participant systems (Thaka, TantaArabNLP, Abjad AI, and Eraserhead) outperform the baselines across evaluation settings. In particular, these systems reduce WER from 31.84% (best baseline) to approximately 23–25% and DER from 9.91% to around 6–8%, demonstrating substantial gains.

A clear performance gap is observed between the text-only baseline and models incorporating ASR outputs, highlighting the importance of speech information for improving diacritization. Furthermore, fine-tuning yields notable improvements over the non-adapted speech+text baseline, especially in DER and WER, indicating better generalization to the target domain.

Across all systems, removing case endings significantly reduces error rates, confirming that syntactic diacritics remain one of the most challenging aspects of Arabic diacritization. Similarly, excluding no-diacritic characters leads to improved scores, suggesting that systems struggle to correctly predict when diacritics should be absent, in addition to predicting their correct forms.

7. Limitations

While KSAA-2026 represents a meaningful step toward multimodal Arabic diacritization, several limitations should be acknowledged. First, the dataset is relatively small in scale, comprising approximately five hours of speech, which may limit the generalizability of trained models to broader real-world conditions. Second, although the dataset covers nine nationalities and includes both Modern Standard Arabic and dialectal speech, the dialectal coverage remains uneven, with some varieties underrepresented relative to their speaker populations across the Arab world. Third, the data contribution track relied on cross-team peer review rather than expert linguistic annotation, which, while practical and scalable, may introduce inconsistencies in quality assessment. Fourth, the evaluation framework focuses on character- and word-level diacritic accuracy but does not capture downstream task performance, such as text-to-speech naturalness or reading comprehension support. Finally, the shared task setting constrains participants to the officially released training data, which limits exploration of transfer learning from larger external resources and may not reflect the full potential of current modeling approaches.

8. Conclusion

KSAA-2026 shared task presented a benchmark for Arabic speech diacritization, highlighting the challenges of combining speech and text for accurate diacritic prediction. The results show that leveraging acoustic information and fine-tuning significantly improves performance over baseline systems.

The diversity of submitted approaches demonstrates the potential for further advancements in multimodal Arabic NLP, while the data contribution track supports the continuous expansion of Arabic speech resources.

9. Ethics Statement

All speech data collected as part of KSAA-2026 was obtained with the informed consent of participants, who were made aware of how their recordings would be used for research purposes. Participation was voluntary, and no personally identifiable information beyond speaker nationality and gender was collected. Recordings were made using scripted prompts covering publicly available text domains, and the dataset is intended solely for non-commercial academic research. Speaker diversity across nationalities and dialects was intentionally incorporated to promote representational fairness. We acknowledge that diacritization systems may carry unintended biases related to dialect or speaking style, and encourage future work to assess and mitigate such biases.

10. References

Alyafeai, Z. and Ahmad, I. (2023). Ashaar: Automatic analysis and generation of Arabic poetry using deep learning approaches. arXiv:2307.06218. doi:10.48550/arXiv.2307.06218.

Elgamal, M., Torki, M., and Hussein, A. (2024). Arabic diacritics in the wild: Exploiting opportunities for improved diacritization. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL).

Shatnawi, A., Alqahtani, S., and Aldarmaki, H. (2024). Automatic restoration of diacritics for speech data sets. In Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL).