Context-Aware Whisper for Arabic ASR Under Linguistic Varieties

Anonymous ACL submission

Abstract

Low-resource ASR remains a challenging problem, especially for languages like Arabic that exhibit wide dialectal variation and limited labeled data. We propose context-aware prompting strategies to adapt OpenAI's Whisper for Arabic speech recognition without retraining. Our methods include decoder prompting with first-pass transcriptions or retrieved utterances, and encoder prefixing using speech synthesized in the target speaker's voice. We introduce techniques such as prompt reordering, speaker-aware prefix synthesis, and modality-specific retrieval (lexical, semantic, acoustic) to improve transcription in real-world, zero-shot settings. Evaluated on nine Arabic linguistic conditions, our approach reduces WER by up to 22.3% on Modern Standard Arabic and 9.2% on dialectal speech, significantly mitigating hallucinations and speaker mismatch.

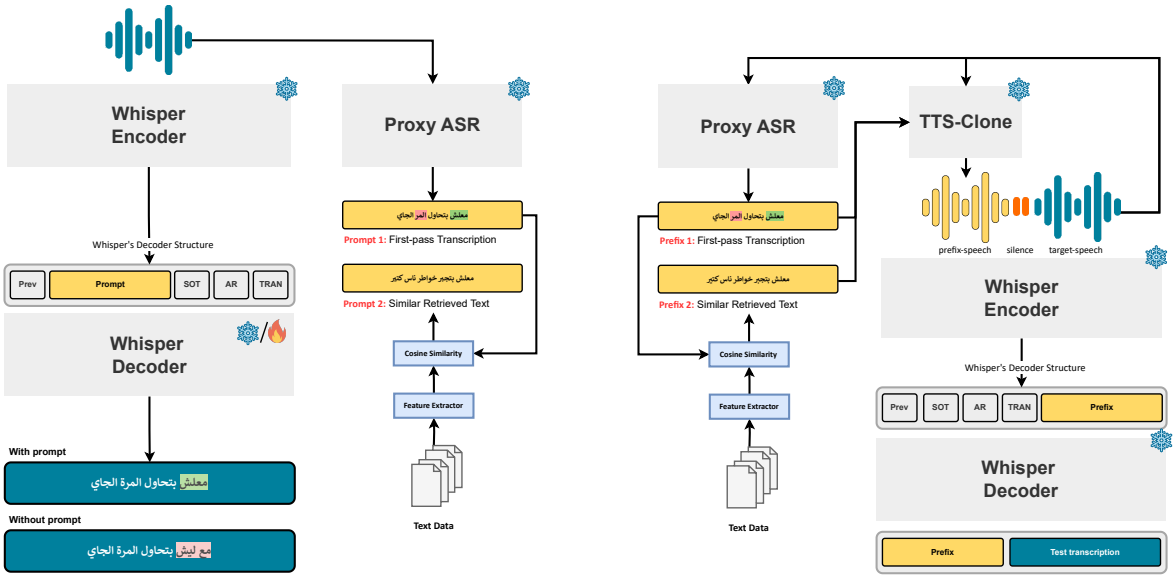
1 Introduction

Recent advances in automatic speech recognition (ASR), especially those powered by large-scale multilingual models, have significantly improved performance across high-resource languages (Pratap et al., 2023; Babu and et al., 2021). Among these, OpenAI's Whisper (Radford et al., 2023a) has shown strong results in languages such as English and Modern Standard Arabic (MSA) (Abdelali et al., 2023). However, its performance on dialectal Arabic remains significantly lower (Team et al., 2025; Talafha et al., 2024), reflecting a persistent challenge in adapting ASR systems to the linguistic diversity of Arabic. This performance gap stems from phonological, lexical, and syntactic differences between MSA and regional dialects (Ali et al., 2016a), which are further exacerbated by the scarcity of annotated data for many dialects. Collecting labeled speech data for each variety is often infeasible due to cost and scalability. As a result, zero-shot ASR on dialectal Arabic continues to yield high error rates, as shown in evaluations on the Casablanca corpus (Talafha et al., 2024).

In this work, we propose a lightweight, context-aware adaptation framework for Whisper that improves its performance on Arabic dialects without any model retraining or architectural changes. Our approach leverages external context, either in the form of first-pass transcriptions or retrieved relevant utterances, as decoder prompts or encoder-attached prefixes. These cues offer valuable lexical and topical signals that help guide Whisper's decoder toward more accurate transcriptions. We also explore the impact of reordering and speaker-matched synthesis to enhance robustness in multi-speaker and informal speech scenarios. We evaluate our approach across a range of dialectal and MSA datasets in zero-shot settings. Our context-aware Whisper consistently achieves lower word error rates (WERs) compared to Whisper and SeamlessM4T baselines. On average, we observe a 9.15% WER reduction on dialectal Arabic, 22.29% on MSA, and 20.54% on accented MSA. These results demonstrate the potential of contextual adaptation as a practical solution for improving ASR on underrepresented Arabic varieties.

2 Related Work

The persistent performance gap between ASR on high-resource languages and low-resource dialects has motivated a variety of adaptation strategies. Several studies highlight that even state-of-the-art models like Whisper and SeamlessM4T perform poorly in zero-shot dialectal settings (Talafha et al., 2024; Abdelali et al., 2023). For example, Whisper often produces hallucinated or repetitive outputs when decoding unseen dialects (Talafha et al., 2023). Attempts to address this using distillation, such as uDistil-Whisper (Waheed et al.,



(a) Prompt-based adaptation. We inject context into Whisper's decoder using either (`Prompt1`) a prior transcription or (`Prompt2`) a retrieved similar sentence. This lightweight approach enhances zero-shot transcription quality, especially for dialectal Arabic.

(b) Prefix-based context integration. We retrieve similar (audio, text) pairs from a reference set. The text is prepended to the decoder, and the corresponding audio, retrieved or voice-cloned, is prepended to the encoder, providing Whisper with aligned acoustic and linguistic context.

Figure 1: Context-aware adaptation strategies: (A) Prompt-based, (B) Prefix-based. We experiment with multiple feature extraction methods and compare each method's performance (see Section 3.1.2). The decoder inputs follow Whisper's multitask training format and include: **Prev:** previous text tokens, **SOT:** start of transcript, **AR:** language tag set to Arabic, and **TRAN:** transcription mode tag. These tokens configure Whisper's decoding behavior and enable contextual prompting.

2024a), are limited by their reliance on pseudo-labels generated by models that already underperform in this context. Recently, prompting-based approaches have emerged as a powerful alternative to fine-tuning. Suh et al. (2024) show that injecting manually written or automatically generated prompts into Whisper's decoder input improves transcription of domain-specific content. Complementarily, Wang et al. (2024b) propose Speech-based In-Context Learning (SICL), which adapts Whisper at inference time by concatenating a few support examples to the encoder input and prepending their transcripts as prefixes. This method achieves over 30% relative WER reduction on unseen Chinese dialects using a k-nearest neighbor retrieval mechanism. These approaches enable test-time adaptation without any gradient updates, making them ideal for low-resource and multilingual scenarios. While these works focus primarily on English and Chinese, the core ideas of leveraging textual or audio context are directly applicable to Arabic, where context-aware prompting can help address the challenges of dialectal variation and data scarcity in ASR. Our work builds on this foundation and adapts these techniques to the Arabic language, introducing novel strategies such as prompt reordering, voice-cloned prefix synthesis, and modality-specific retrieval to enhance transcription quality in real-world, zero-shot scenarios.

3 Methodology

Our approach leverages proxy transcriptions from an auxiliary ASR system to guide Whisper's decoding. Specifically, we use SM4T (Barrault et al., 2023) to generate first-pass hypotheses that serve as contextual prompts or prefixes. Although SM4T is not designed for prompt-based decoding, its high-quality ASR outputs offer lightweight, plug-in contextual cues that improve Whisper's recognition, especially in dialectal and low-resource settings, without requiring joint training or architectural changes.

We build upon Whisper (Radford et al., 2023b), a multilingual encoder-decoder Transformer (Vaswani et al., 2017) model for ASR. Whisper consists of an audio encoder and an autoregressive

token-based decoder. The encoder takes a log-Mel spectrogram of the input audio and produces a sequence of latent audio representations, which are then decoded into text tokens by the decoder. Whisper is trained on a large-scale collection of diverse audio-text pairs and is known for its robust performance across 98 languages covering different domains. Whisper's decoding process is autoregressive and conditioned on a sequence of special tokens and optional user-provided prompts. As illustrated in Figure1, the decoder input typically follows this structure: `|PREV|` → `[prompt tokens]` → `|SOT|` → `|lang|` → `|TASK|` → `[output]`. The `|PREV|` token marks the beginning of the prompt section, followed by the prompt tokens, which can include lexical or semantic information related to the target utterance. The token `|SOT|` signals the beginning of the expected output, followed by language and task specification tags (e.g., `<|ar|>` and `<|transcribe|>` for Arabic transcription). We explore **context-aware decoding** by leveraging Whisper's support for decoder prompts under two strategies: prompt-based and prefix-based context integration.

3.1 Prompt-based Methods

Here, we explore injecting the transcribed or retrieved textual context into Whisper's decoder directly after the `|PREV|` token to guide the transcription of dialectal words. We explore two types of prompts: the first-pass transcription of the target audio and a semantically similar retrieved text. These configurations are illustrated in Figure 1a as *Prompt1* and *Prompt2*, respectively.

3.1.1 First-Pass Transcription as Prompt

We use first-pass transcriptions generated by SM4T[1] (Barrault et al., 2023) as contextual prompts for Whisper. Our choice of SM4T is motivated by the findings of Waheed et al. (2024a), which reported state-of-the-art zero-shot performance across nearly all Arabic dialects. We hypothesize that providing recognized dialectal words as prompts can guide Whisper toward outputting more accurate and dialect-aware transcriptions. Formally, the model generates the output sequence $\hat{y}$ autoregressively as:

$$\hat{y} = \arg\max_{y} \prod_{t=1}^{T} P(y_t \mid y_{<t}, \mathbf{x}, \mathbf{p}; \theta)$$

where $\mathbf{x}$ is the input audio, $\mathbf{p}$ is the textual prompt (e.g., SM4T output), y_t is the token at time step t, and θ are the model parameters.

3.1.2 Retrieved Similar Text as Prompt

A limitation of the first-pass approach is its reliance on the accuracy of the ASR model used to generate first-pass transcriptions (i.e., SeamlessM4T); if the model misrecognizes a dialectal word, the error may propagate to Whisper's final output. To address this, we propose retrieving a *similar* sentence from a large, human-written text corpus as an alternative prompt source.

In this approach, we retrieve similar sentences from a large textual corpus. In our experiments, we use a 500K ASR speech-transcription dataset (Waheed et al., 2024a). Since the retrieval operates purely on text, it is speech-independent, an important advantage in low-resource settings where large text corpora are more readily available than labeled speech data. We define a sentence as *similar* if it has high lexical or semantic overlap with the reference first-pass transcription $\mathbf{t}_{text}$, following a rationale similar to WER and CER computation. We evaluate two models to produce these transcriptions: (a) a character-level ASR (MMS (Pratap et al., 2024)) and (b) a subword-level ASR (SM4T). As shown in the "feature extractor" block in Figure 1a, we experiment with four similarity metrics to retrieve the most relevant candidate $\mathbf{p} \in \mathcal{C}$ for each test utterance. These include (1) lexical features based on character-level TF-IDF (Piskorski and Jacquet, 2020), (2) semantic features based on sentence embeddings via SentenceTransformer (Reimers and Gurevych, 2019), (3) speech embeddings derived from Whisper's encoder (Radford et al., 2023a), and (4) speaker embeddings from ECAPA-TDNN (Desplanques et al., 2020). For each test utterance with audio input $\mathbf{x}$, we first obtain a first-pass transcription $\mathbf{t}_{text}$, which is embedded using a feature extractor $f(\cdot)$. We then embed all candidate sentences $\mathbf{s}_i \in \mathcal{C}$ from a reference corpus using the same extractor and compute cosine similarity:

$$\mathbf{p} = \arg\max_{\mathbf{s}_i \in \mathcal{C}} \cos\left(f(\mathbf{t}_{text}), f(\mathbf{s}_i)\right)$$

The retrieved sentence $\mathbf{p}$ is then used as a contextual prompt injected into Whisper's decoder input

[1] https://huggingface.co/facebook/seamless-m4t-v2-large

to guide the transcription of $\mathbf{x}$. The output sequence $\hat{y}$ is generated autoregressively as:

$$\hat{y} = \arg\max_{y} \prod_{t=1}^{T} P(y_t \mid y_{<t}, \mathbf{x}, \mathbf{p}; \theta)$$

3.2 Prefix-based Methods

In contrast to the prompt-based approach, which injects contextual information solely into the decoder, prefix-based methods augment both the encoder and decoder inputs of Whisper with contextual $(audio, text)$ pairs. These pairs are selected to be semantically similar to the target utterance and are prepended to the input stream, thereby enabling Whisper to perform context-aware transcription across the full encoder-decoder pipeline. This approach builds on the Speech-based In-Context Learning (SICL) framework introduced by Wang et al. (2024b), which demonstrates that Whisper can adapt to new dialects and speakers by conditioning on a few relevant speech-text examples at test time, without requiring any updates to the model parameters. Figure 1b illustrates the overall setup. Given a test utterance with audio input $\mathbf{x}$, we retrieve a similar $(\mathbf{x}_{ctx}, \mathbf{p})$ pair from a reference dataset, where $\mathbf{x}_{ctx}$ is the context audio and $\mathbf{p}$ is the corresponding transcript. Following the SICL paradigm, we concatenate $\mathbf{x}_{ctx}$ and $\mathbf{x}$, separated by a 1-second silence, and feed the resulting sequence into the encoder. The text prefix $\mathbf{p}$ is prepended to the decoder input. This paired context enables Whisper to leverage both acoustic and linguistic cues during generation. The final output sequence $\hat{y}$ is generated autoregressively as:

$$\hat{y} = \arg\max_{y} \prod_{t=1}^{T} P(y_t \mid y_{<t}, \mathbf{x}_{ctx} \oplus \mathbf{x}, \mathbf{p}; \theta)$$

where $\mathbf{x}_{ctx} \oplus \mathbf{x}$ denotes the concatenation of the context and test audio, and $\mathbf{p}$ is the text prefix used to guide the decoder.

3.2.1 Retrieval-based Prefixing

We begin by identifying a semantically similar training example using character-level TF-IDF similarity over transcriptions, which we found to outperform other retrieval methods in this context (see Section 5.2). The retrieved speech $\mathbf{x}_{ctx}$ and its corresponding transcript $\mathbf{p}$ are then prepended to the test utterance and used as contextual inputs to Whisper's encoder and decoder, respectively.

In the work by Wang et al. (2024b), the dataset features repeated speakers across training and test utterances, making the concatenation of $\mathbf{x}_{ctx} \oplus \mathbf{x}$ more seamless and coherent, often resembling a single extended utterance. This aligns well with Whisper's design, which assumes single-speaker input[2]. However, in our setting, the context and test utterances often come from different speakers. We observe that this speaker mismatch can result in inconsistent behavior, with Whisper frequently ignoring one of the speakers or producing fragmented outputs. For example, in the absence of speaker alignment, Whisper may truncate the first utterance or hallucinate speaker turns. This issue of speaker mismatching is addressed in the next section.

3.2.2 Retrieval-based Prefixing with Voice Cloning

To address the speaker mismatch issue discussed earlier, we synthesize the contextual audio using a **cloned voice** that matches the speaker identity of the target test utterance. Specifically, we take the retrieved transcription $\mathbf{p}$ and synthesize a new contextual audio signal $\tilde{\mathbf{x}}_{clone}$ using a TTS model (XTTS (Casanova et al., 2024)) conditioned on the speaker embedding extracted from the test audio $\mathbf{x}$. This results in a speaker-consistent input that Whisper perceives as originating from a single speaker. Formally, we model the synthesized contextual audio $\tilde{\mathbf{x}}_{clone}$ as:

$$\tilde{\mathbf{x}}_{clone} = TTS(\mathbf{p},\, SPK(\mathbf{x}))$$

where $\mathbf{p}$ is the retrieved text prompt, $\mathbf{x}$ is the test utterance audio, $SPK(\mathbf{x})$ extracts the speaker embedding from $\mathbf{x}$, and TTS is conditioned on both the text and the target speaker identity. This approach not only aligns the speaker characteristics of the context and test segments but also removes the need for parallel speech-text data, an important advantage in low-resource and dialectal settings where such data is often scarce. By enabling Whisper to process a seamless input with unified acoustic characteristics, this method enhances both transcription accuracy and inclusivity. Formally, the input to Whisper becomes the concatenated audio $\tilde{\mathbf{x}}_{clone} \oplus \mathbf{x}$, with $\mathbf{p}$ serving as the corresponding decoder prefix.

3.3 First-Pass Transcription Prefixing with Voice Cloning

Instead of retrieving external examples, this method constructs the prefix directly from the test

[2] https://github.com/openai/whisper/discussions/434

utterance itself. We begin by transcribing the test audio $\mathbf{x}$ to obtain a first-pass transcription $\mathbf{t}_{text}$, as described in Section 3.1.1. We then synthesize the corresponding audio using a voice cloned from the same test utterance. As in the retrieval-based prefixing method, the synthesized audio $\tilde{\mathbf{x}}_{clone}$, followed by a 1-second silence and the test audio $\mathbf{x}$, is fed into Whisper's encoder. The corresponding text $\mathbf{t}_{text}$ is used as the decoder prefix.

4 Experiments

In this section, we evaluate our proposed methods under varying linguistic conditions, including *Modern Standard Arabic (MSA)*, *Accented MSA*, and both *external* and *internal dialectal* datasets.

4.1 Datasets

Common Voice 15.0 (CV15). A crowd-sourced dataset of read Arabic speech (Ardila et al., 2019). Utterances written in *MSA*, the formal variety used widely across the Arab world in news broadcasts, education, and official contexts.

MGB-2/3/5. This collection comes from the Arabic Multi-Genre Broadcast (MGB) challenges (Ali et al., 2016b, 2017, 2019), which feature speech from real-world broadcast content. MGB-2 (around 1,200 hours) contains *MSA* with other dialects mixed in. MGB-3 (≈6 hours) focuses on *Egyptian dialect*, while MGB-5 (≈6 hours) focuses on *Moroccan Arabic*. We present MGB-3 and MGB-5 as *external dialectal data*. We manually validated MGB samples and found errors like omissions, mismatches, and typos.

FLEURS. The Arabic portion of FLEURS (Conneau et al., 2023). Features read speech sourced from news and web content. The speech is in *MSA* but spoken with an Egyptian accent, as known as *accented MSA* (Waheed et al., 2024b; Talafha et al., 2023).

In-House Dialectal Sets. This group includes five conversational test sets: *Algerian*, *Jordanian*, *Palestinian*, *Emirati (UAE)*, and *Yemeni*. Each consists of multi-speaker dialogue recordings in regional dialects, collected and manually annotated and validated. We present these as *internal dialectal data*.

4.2 Baseline Models

Table 1 presents WER/CER for all dialectal variations. All error rates are measured after performing preprocessing on the reference and prediction texts. See Appendix A.1. We kept all models on their default settings. See model details in Appendix A.2. We begin with baselines: zero-shot Whisper-large-v3[3] and SM4T[4]. In line with the experiments done by Waheed et al. (2024a), SM4T yields lower error rates than Whisper on most dialectal sets in our experiments. Overall, both Whisper and SM4T perform better on MSA ($\approx 15.79\%$ and $\approx 14.24\%$, respectively) than on dialects ($\approx 57.85\%$ and $\approx 57.48\%$, respectively), illustrating the large gap between MSA and dialectal ASR.

4.3 MSA (CV15, MGB-2)

For MSA, although Whisper is already relatively strong (WER 15.55% on CV15, 16.02% on MGB-2), prompt-based methods further improve MSA accuracy. For instance, prompting with the SM4T transcription reduces CV15 error rates to (10.40/3.18), a roughly 33% relative reduction. However, on MGB-2, we observe a surprising degradation: WER spiked to 47.61%. Upon analyzing selected samples, we identify several consistent failure modes: completions (i.e., Whisper attempting to continue the prompt), empty transcriptions, and hallucinated phrases. These behaviors reflect inherent properties of autoregressive decoding where Whisper generates text token by token, which can lead it to overfit on the prompt and treat it as prior context to be continued. We noticed that randomly shuffling the prompt words sharply reduces this behavior in MGB-2, bringing the WER to 15.01, overcoming vanilla Whisper. Changing the order of the prompt disrupts Whisper's tendency to depend on the prompt as a coherent sequence and perceives it as a bag of words instead. To better understand the impact of reversed prompting on hallucination reduction, we manually analyzed 30 samples from the development set where sentence-level WER dropped from ≥ 1 to 0 when using the reversed prompt. We found that hallucinations typically occurred in cases of incomplete utterances (16 samples), background music (4), simultaneous interpretation or voice-over (4), and multi-speaker dialogue (6). In many cases, Whisper hallucinated generic filler content (e.g., اشتركوا في القناة or ترجمة نانسي قنقر), which trans-

[3] https://huggingface.co/openai/whisper-large-v3

[4] https://huggingface.co/facebook/seamless-m4t-v2-large

Language Condition		Baselines		**Ours:** Prompt-based (←)					**Ours:** Prefix-based (→)		
		SM4T	W-v3	W←FPT	W←Rand	W←Rev	W←SMms	W←SSea	W→SSea No-Clone	W→SSea Clone	W→Pt Clone
MSA											
CV15.0	*WER/*	11.12/	15.55/	**10.40 /**	12.01/	12.12 /	13.69/	12.69/	15.67/	**11.26/**	**10.28/**
	CER	3.55	5.06	**3.18**	3.93	3.88	4.59	4.29	7.45	**3.46**	**3.29**
MGB2	*WER/*	17.35/	16.02/	47.61/	16.7/	**15.01/**	17.28/	16.51/	17.15/	**14.66/**	**14.26/**
	CER	8.73	7.64	36.61	9.09	**7.66**	7.48	7.24	7.94	**5.97**	**6.36**
Avg MSA	*WER/*	14.24/	15.79/	29.01/	14.36/	**13.57/**	**15.49/**	**14.60/**	16.41/	**12.96/**	**12.27/**
	CER	06.14	06.35	19.90	06.51	**05.77**	**06.04**	**05.77**	07.70	**04.72**	**04.83**
Accented MSA											
Fleurs	*WER/*	7.66/	9.2/	17.34/	**7.36/**	**7.31/**	12.18/	12.21/	11.56/	10.22/	9.31/
	CER	4.0	2.77	12.56	**3.73**	**3.76**	3.93	4.34	4.69	3.60	**2.72**
External Dialects											
MGB3	*WER/*	31.48/	35.9/	63.47/	31.62/	**31.14/**	37.15/	35.45/	35.45/	34.22/	33.51/
	CER	15.75	17.67	50.14	15.98	**15.26**	18.43	17.47	17.22	15.90	18.64
MGB5	*WER/*	77.43/	79.16/	**76.04/**	**69.37/**	**69.89/**	**76.90/**	**75.23/**	**74.35/**	**73.70/**	**68.21/**
	CER	43.62	45.1	**45.66**	**35.55**	**35.49**	**42.02**	**40.37**	**38.90**	**37.13**	**33.96**
Avg Ext	*WER/*	54.46/	57.53/	69.76/	**50.50/**	**50.52/**	57.03/	55.34/	**54.90/**	**53.96/**	**50.86/**
	CER	29.69	31.39	47.90	**25.77**	**25.38**	30.23	28.92	**28.06**	**26.52**	**26.30**
Internal Dialects											
ALG	*WER/*	86.89/	78.6/	**77.83/**	**73.07/**	**73.13/**	**76.59/**	**74.68/**	**76.87/**	**74.53/**	**70.08/**
	CER	45.5	37.81	**39.19**	**31.26**	**30.38**	**34.70**	**34.17**	**36.57**	**31.28**	**31.85**
JOR	*WER/*	38.29/	40.79/	**37.34/**	**37.52/**	**37.34/**	**38.01/**	**35.90/**	**36.96/**	**35.00/**	**36.04/**
	CER	12.01	13.55	**12.12**	**12.25**	**12.12**	**13.39**	**12.59**	**13.61**	**11.79**	**14.10**
PAL	*WER/*	48.82/	50.38/	**46.12/**	**46.55/**	**46.12/**	**45.28/**	**45.24/**	**44.38/**	**42.94/**	**52.78/**
	CER	16.49	17.52	**14.98**	**14.9**	**14.96**	**16.48**	**16.69**	**16.65**	**14.92**	**27.16**
UAE	*WER/*	51.79/	55.03/	**49.1/**	**49.45/**	**48.98/**	**50.22/**	**48.32/**	**51.02/**	**47.90/**	**51.91/**
	CER	19.75	22.98	**18.13**	**18.48**	**18.05**	**21.01**	**20.06**	**23.26**	**19.03**	**24.53**
YEM	*WER/*	70.22/	62.51/	**60.74/**	**60.35/**	**60.21/**	64.82/	62.60/	63.32/	**60.73/**	64.70/
	CER	28.97	24.42	**23.49**	**22.97**	**23.28**	26.51	25.47	27.30	**23.01**	30.56
Avg Int	*WER/*	59.20/	57.46/	**54.23/**	**53.39/**	**53.16/**	**54.98/**	**53.35/**	**54.51/**	**52.22/**	**55.10/**
	CER	24.54	23.26	**21.58**	**19.97**	**19.76**	**22.42**	**21.80**	**23.48**	**20.01**	**25.64**
Avg All	*WER/*	44.11/	44.31/	48.60/	**40.40/**	**40.13/**	**43.21/**	**41.88/**	**42.67/**	**40.52/**	**41.11/**
	CER	19.84	19.45	25.61	**16.81**	**16.48**	**18.85**	**18.27**	**19.36**	**16.61**	**19.32**
Avg Dia	*WER/*	57.85/	57.48/	58.66/	**52.56/**	**52.40/**	**54.98/**	**53.35/**	**54.51/**	**52.22/**	**55.10/**
	CER	26.01	25.58	29.10	**21.63**	**21.36**	**22.42**	**21.80**	**23.48**	**20.01**	**25.64**

Table 1: WER (↓) and CER (↓) across various Arabic speech conditions using baseline and context-aware Whisper decoding strategies. Baseline models are **SM4T:** SM4T and **W-v3:** Whisper-large-v3. Our prompt-based methods (←) inject contextual text into the decoder using **W←FPT:** first-pass transcriptions. ←**Rand:** randomly shuffling the prompt's words and ←**Rev:** reversing the prompt word's order. ←**SMms** and ←**SSea:** retrieving similar sentences based on MMS or SM4T, respectively. Prefix-based methods (→) concatenate contextual (speech, text) pairs at the encoder/decoder inputs. **No-Clone:** retrieve the speech for that similar example. **Clone:** Use TTS to clone the speech for that similar example based on the target utterance.

late to "subscribe to the channel" and "Translated by Nancy Kangar", respectively. It seemingly attempted to 'complete' utterances it interpreted as finished. These behaviors are likely inherited from Whisper's training data, which includes YouTube videos and subtitles, where endings often feature music or silence. Similar hallucinations were observed across other languages, such as *Untertitel im Auftrag des ZDF, 2017* in German[5], *Tekstet av Nicolai Winther* in Norwegian[6], or generic tags like *[applause]* in English[7]. An example of an incomplete utterance is: وايضا اعطى للعملية برمتها نوع من الـ ("It also gave the whole process a kind of"). This sentence ends with a rising intonation and lacks a complete semantic conclusion, making it a likely candidate for hallucinated completions. Table 2 shows some examples of prompt-completion or generic-hallucination cases.

For CV15, both shuffling and reversing slightly increase error rates to (12.02/3.93) and (12.12/3.88), respectively, in comparison with only using the normal prompt, but still better than vanilla Whisper.

[5] https://gist.github.com/riotbib/3b3c5f817b55b68801d14b8bdb02df09

[6] https://medium.com/@lehandreassen/who-is-nicolai-winther-985409568201

[7] https://github.com/openai/whisper/discussions/2608

Prompting using similar text also helps MSA modestly; using similar text from data source (i.e., 500K) based on MMS as prompt yields a WER of $\approx 15.5\%$ on average MSA (from vanilla Whisper at 15.8%). Furthermore, retrieving similar text from the same dataset based on SM4T transcription yields an even lower WER of $\approx 14.6\%$. This similarity-based approach provides the decoder with salient words that guide whisper's decoding process and aligns with the way upon which the model (i.e., Whisper) was trained.

We observe that prefix-based approaches also improve Whisper's performance on MSA, but only when speakers' characteristics are similar. When prefixing with a retrieved context utterance without speaker adaptation[8], performance on MSA remains nearly unchanged or degrades slightly: on CV15, error rates at (15.67/7.45), compared to Whisper's baseline (15.55/5.06), whereas on MGB-2, error rates increased to (17.15/7.94). In contrast, prefixing with the *synthesized* similar text in the target speaker's voice substantially improves results, reducing error rates to (11.26/3.46) on CV15 and (14.66/5.97) on MGB-2, bringing the average MSA WER down to 12.96. The best performance is achieved when the cloned prefix uses the synthesized SM4T transcription as a prefix, which yields (10.28/3.29) on CV15 and (14.26/6.36) on MGB-2, reducing the average MSA WER to 12.27.

4.4 Accented MSA (Fleurs)

For Accented MSA, Whisper's baseline is already relatively competitive (9.20/2.77), only slightly behind the stronger SM4T ASR baseline (7.66/4.00). SM4T prompting has mixed effects. We noticed that injecting the transcription *as is* causes similar behavior to MGB-2, yielding (17.34/12.56). However, changing the order of the same prompt, restores and even improves performance: shuffling yields (7.36/3.73), while reversing delivers the best result at (7.31/3.76), a 21% relative WER reduction over the Whisper baseline. This mirrors the prompt-completion pattern in Table 2, where reversing suppresses boilerplate hallucinations such as اشتركوا في القناة. Similar text prompting with TF-IDF hindered the performance by nearly 33%, as Fleurs is out of the textual corpus domain.
In the prefix approach, without speaker adaptation, performance degrades to (11.56/4.69) compared with the baselines. Synthesising the same text in a *cloned* voice aligned to the target speaker yields slightly better results at (10.22/3.60), and prefixing the cloned SM4T transcription slightly better at (9.31/2.72), but both approaches still do not even outperform the baselines. This shows that the *prompt reversal* continues to be the most effective method for Accented MSA.

4.5 External Dialectal Datasets (MGB-3, MGB-5)

Baseline Whisper performs poorly on dialectal benchmarks, with 35.90% WER on MGB-3 and 79.16% on MGB-5. Naively prompting with SM4T transcriptions harms MGB-3, spiking WER to 63.47%, as Whisper often treats the prompt as ground truth and returns empty or hallucinated outputs. On MGB-5, however, this yields a modest 4% relative improvement (76.04% WER). Reordering prompt words mitigates these issues. Shuffling reduces WER to 31.62% on MGB-3 and 69.37% on MGB-5. Reversing performs best on MGB-3 (31.14/15.26) and matches shuffling on MGB-5 (69.89/35.49), confirming that disrupting prompt syntax discourages premature decoding. Similar-text prompting shows limited gains. MMS-based retrieval slightly worsens MGB-3 (37.15/18.43) and marginally improves MGB-5 (76.90/42.02). Using SM4T narrows this gap but remains less effective than prompt reordering. Prefixing proves more stable. Concatenating raw similar examples yields modest gains (MGB-5 at 74.35/38.90), but speaker mismatch sometimes leads to dropped content. Voice cloning improves consistency (34.22/15.90 on MGB-3, 73.70/37.13 on MGB-5). Using cloned SM4T transcriptions performs best on MGB-5 (68.21/33.96), cutting WER by nearly 14% relative. Averaged across datasets, prefixing with cloned SM4T achieves 50.86/26.30. Still, prompt reordering—shuffling (50.50/25.77) and reversing (50.52/25.38), delivers the best overall performance, improving WER by 12% and CER by 19% over baseline for dialectal ASR.

4.6 Internal Dialectal datasets

Across the five internal dialectal datasets, which include Algerian from North Africa, Jordanian and Palestinian from the South Levant, Emirati from the Gulf, and Yemeni from the southern Arabian Peninsula, the Whisper baseline averages

[8] We define speaker adaptation as the process of cloning the target speaker's voice from the target audio in hand (i.e., test example) and using it to synthesize the prefix.

(57.46/23.26), which is significantly worse than on MSA at (15.79/06.35). Error rates vary substantially between dialects, with Algerian showing the highest (78.60/37.81) and Jordanian the lowest (40.79/13.55).

Prompting with the SM4T transcription narrows the average to (54.23/21.58), driven mainly by gains in the South Levant dialects (i.e., (37.34/12.12) for Jordanian and (46.12/14.98) for Palestinian as well as Emirati at (49.1/18.13)), while Algerian and Yemeni showed small gains at (77.83/39.19) and (60.74/23.49), respectively. Shuffling the prompt was more effective for Algerian as it lowered error rates to (73.07/31.26), but it did not show significant changes for the other dialects compared to the normal-ordered prompt. Although shuffling lowers the mean to (53.39/19.97), reversing yielded a slightly better average of (53.16/19.76), outperforming the baseline WER by just under 7.5%.

Similar text prompting gives a modest boost when the reference text comes from the same SM4T pipeline (53.35/21.80) and less when drawn from MMS (54.98/22.42); Jordanian falls to (35.90/12.59) but Algerian scarcely budges, implying lexical overlap (i.e., between MSA and South Levant dialects) drives the benefit rather than similarity.

Moving to prefix-based methods, concatenating raw similar audio without speaker adaptation yields (54.51/23.48), as it causes Whisper to experience the aforementioned multi-speaker issue where it ignores one of the speakers, with failures most visible in Yemeni (63.32/27.30). Injecting a voice-cloned retrieval prefix aligns speaker characteristics and produces the best overall scores, averaging (52.22/20.01), outperforming the baseline by around 9%; Jordanian improves to (35.00/11.79), Palestinian to (42.94/14.92), Emirati to (47.90/19.03), and even Algerian drops to (74.53/31.28). And while using a cloned SM4T prefix degraded in almost all dialects and caused the overall error rates to rise to (55.10/25.64), Algerian achieved its best performance at (70.08/31.85), an improvement of 10.4% relative.

5 Discussion

5.1 Impact of TTS on Prefix-Based Decoding

In our prefix-based methods, we rely on synthetic speech to provide Whisper with context. A crucial concern was whether the quality of the synthesized speech would degrade recognition accuracy. To assess this, we compared Whisper's performance when using real versus TTS-generated context across three datasets (CV15, MGB3, FLEURS), as shown in Figure 2. Despite minor increases in WER (e.g., +6.79% on CV15, +4.98% on MGB-3, and +1.05% on Fleurs), the impact was modest, with average degradation across datasets remaining within 4.27% WER and 3.41% CER, indicating that synthesized audio did not significantly alter the decoding behavior of Whisper. These findings suggest that our TTS pipeline, when combined with speaker voice cloning, preserves sufficient acoustic fidelity to act as an effective alternative to labeled speech utterances.

5.2 Feature Extractor Design and TF-IDF Effectiveness

For text-based prompt retrieval, we evaluated four types of feature extractors as means to measure similarity, TF-IDF, text embeddings, speech embeddings, and speaker embeddings, using WER/CER as the primary indicators. Among these, character-level TF-IDF consistently outperformed other methods, reducing WER from 22.84% (vanilla) to 17.89%, outperforming dense text embeddings (20.04%), speech-based embeddings (24.78%), and speaker-based embeddings (27.16%) as explained in Table 3 and Appendix **??**. In addition, TF-IDF's independence from speech input makes it particularly suitable for low-resource scenarios (i.e., Dialectal Arabic ASR), as it relies only on textual corpora, which are more widely available than labeled audio. This further motivated our use of TF-IDF as the default retrieval method in all experiments.

6 Conclusion

In this paper, we explored context-aware decoding strategies to improve Whisper's performance on dialectal and accented Arabic speech. Our two complementary methods, *prompt-based*, which inject contextual text into the decoder, and *prefix-based*, which prepend contextual speech and text, show consistent gains. All experiments were conducted in a zero-shot setting, as our goal is to enhance ASR in low-resource conditions without modifying the model architecture. In future work, we plan to explore prompted fine-tuning, support for code-switching, and unified approaches that combine prompting and prefixing within a single context-aware framework.

Limitations

Despite the consistent improvements achieved by our context-aware strategies, several limitations must be acknowledged. These span computational trade-offs, model constraints, and coverage gaps, which may affect real-world applicability and generalizability.

Computational and Latency Overhead: Our methods introduce additional processing steps per utterance, such as proxy ASR for first-pass hypotheses, feature extraction for retrieval, or TTS synthesis for prefix construction, which increase computational demands. These steps also add latency and cost, making real-time or edge deployment more challenging. Prior work integrating retrieval or kNN with Whisper similarly reports increased decoding overhead (Wang et al., 2024a; Nachesa and Niculae, 2024; Shen et al., 2025).

Model Error Propagation: The performance of our approach is tied to the quality of auxiliary components. Errors in proxy ASR or TTS (e.g., unnatural prosody, mispronunciations, or lack of dialectal support) can degrade the effectiveness of contextual prompts.

Prompt Length Constraint: Whisper only considers the final 224 tokens of the prompt during decoding[9], limiting the utility of longer contextual inputs. While some implementations use a 448-token window, only half is usable for prompts[10].

Dialectal Coverage Gaps: Although we evaluate across several datasets and dialects, Arabic remains underrepresented in ASR resources. Other dialects, such as Sudanese, Mauritanian, and Iraqi, were not included in our experiments, and existing benchmarks may carry domain, genre, or demographic biases.

Retrieval Limitations: Retrieval quality is influenced by corpus characteristics and retrieval method. Lexical techniques like TF-IDF are sensitive to tokenization and spelling variation, while semantic and acoustic approaches may introduce bias toward certain genres or speaker types. Additionally, large corpora improve recall but incur higher indexing and search-time costs, which can further impact latency and scalability.

Limited Exploration of Prompting Strategies: Our exploration of prompting strategies remains limited. Beyond TF-IDF and basic reordering techniques (e.g., reverse, shuffle), many alternatives remain unexplored. For example, LLM-based prompt generation could be employed to produce domain-aware cues, such as emphasizing dialect-specific keywords, similar to the work of Suh et al. (2024).

[9] https://platform.openai.com/docs/guides/speech-to-text

[10] https://github.com/huggingface/transformers/issues/27445

References

Ahmed Abdelali, Hamdy Mubarak, Shammur Absar Chowdhury, Maram Hasanain, Basel Mousi, Sabri Boughorbel, Yassine El Kheir, Daniel Izham, Fahim Dalvi, Majd Hawasly, and 1 others. 2023. Larabench: Benchmarking arabic ai with large language models. *arXiv preprint arXiv:2305.14982*.

Ahmed Ali, Peter Bell, Mark Gales, Kevin Kilgour, Pierre Lanchantin, Xunying Liu, Steve Renals, and 1 others. 2016a. The mgb-2 challenge: Arabic multi-dialect broadcast media recognition. In *IEEE Spoken Language Technology Workshop (SLT)*, pages 499–504.

Ahmed Ali, Peter Bell, James Glass, Yacine Messaoui, Hamdy Mubarak, Steve Renals, and Yifan Zhang. 2016b. The mgb-2 challenge: Arabic multi-dialect broadcast media recognition. In *2016 IEEE Spoken Language Technology Workshop (SLT)*, pages 279–284. IEEE.

Ahmed Ali, Stephan Vogel, and Steve Renals. 2017. Speech recognition challenge in the wild: Arabic mgb-3. In *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 316–322. IEEE.

Ahmed Ali and 1 others. 2019. The mgb-5 challenge: Arabic multi-dialect broadcast media recognition for youtube videos. In *Proc. ASRU*.

Rosana Ardila, Megan Branson, Kelly Davis, Michael Henretty, Michael Kohler, Josh Meyer, Reuben Morais, Lindsay Saunders, Francis M Tyers, and Gregor Weber. 2019. Common voice: A massively-multilingual speech corpus. *arXiv preprint arXiv:1912.06670*.

Arun Babu and et al. 2021. Xls-r: Self-supervised cross-lingual speech representation learning at scale. In *Proc. Interspeech*.

Loïc Barrault, Yu-An Chung, Mariano Cora Meglioli, David Dale, Ning Dong, Paul-Ambroise Duquenne, Hady Elsahar, Hongyu Gong, Kevin Heffernan, John Hoffman, and 1 others. 2023. Seamlessm4t: massively multilingual & multimodal machine translation. *arXiv preprint arXiv:2308.11596*.

Edresson Casanova, Kelly Davis, Eren Gölge, Görkem Göknar, Iulian Gulea, Logan Hart, Aya Aljafari,

Joshua Meyer, Reuben Morais, Samuel Olayemi, and 1 others. 2024. Xtts: a massively multilingual zero-shot text-to-speech model. *arXiv preprint arXiv:2406.04904*.

Alexis Conneau, Min Ma, Simran Khanuja, Yu Zhang, Vera Axelrod, Siddharth Dalmia, Jason Riesa, Clara Rivera, and Ankur Bapna. 2023. Fleurs: Few-shot learning evaluation of universal representations of speech. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, pages 798–805. IEEE.

Brecht Desplanques, Jenthe Thienpondt, and Kris Demuynck. 2020. Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification. *arXiv preprint arXiv:2005.07143*.

Maya K Nachesa and Vlad Niculae. 2024. knn for whisper and its effect on bias and speaker adaptation. *arXiv preprint arXiv:2410.18850*.

Jakub Piskorski and Guillaume Jacquet. 2020. Tf-idf character n-grams versus word embedding-based models for fine-grained event classification: A preliminary study. In *Proceedings of the Workshop on Automated Extraction of Socio-political Events from News 2020*, pages 26–34.

Vineel Pratap, Andros Tjandra, Bowen Shi, Paden Tomasello, Arun Babu, Sayani Kundu, Ali Elkahky, Zhaoheng Ni, Apoorv Vyas, Maryam Fazel-Zarandi, and 1 others. 2024. Scaling speech technology to 1,000+ languages. *Journal of Machine Learning Research*, 25(97):1–52.

Vineel Pratap, Qiantong Xu, Tatiana Likhomanenko, and 1 others. 2023. Scaling speech technology to 1,000+ languages. *arXiv preprint arXiv:2305.13574*.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023a. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pages 28492–28518. PMLR.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023b. Robust speech recognition via large-scale weak supervision. In *Proceedings of the 40th International Conference on Machine Learning*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Peng Shen, Xugang Lu, and Hisashi Kawai. 2025. Retrieval-augmented speech recognition approach for domain challenges. *arXiv preprint arXiv:2502.15264*.

Jiwon Suh, Injae Na, and Woohwan Jung. 2024. Improving domain-specific asr with llm-generated contextual descriptions. In *Proceedings of Interspeech 2024*. ArXiv:2407.17874.

Bashar Talafha, Karima Kadaoui, Samar Mohamed Magdy, Mariem Habiboullah, Chafei Mohamed Chafei, Ahmed Oumar El-Shangiti, Hiba Zayed, Rahaf Alhamouri, Rwaa Assi, Aisha Alraeesi, and 1 others. 2024. Casablanca: Data and models for multidialectal arabic speech recognition. *arXiv preprint arXiv:2410.04527*.

Bashar Talafha, Abdul Waheed, and Muhammad Abdul-Mageed. 2023. N-shot benchmarking of whisper on diverse arabic speech recognition. *arXiv preprint arXiv:2306.02902*.

Fanar Team, Ummar Abbas, Mohammad Shahmeer Ahmad, Firoj Alam, Enes Altinisik, Ehsannedin Asgari, Yazan Boshmaf, Sabri Boughorbel, Sanjay Chawla, Shammur Chowdhury, and 1 others. 2025. Fanar: An arabic-centric multimodal generative ai platform. *arXiv preprint arXiv:2501.13944*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Abdul Waheed, Karima Kadaoui, Bhiksha Raj, and Muhammad Abdul-Mageed. 2024a. udistil-whisper: Label-free data filtering for knowledge distillation in low-data regimes. *arXiv preprint arXiv:2407.01257*.

Nasser Waheed and 1 others. 2024b. To distill or not to distill? on the robustness of robust knowledge distillation. *arXiv preprint arXiv:2406.04512*.

Siyin Wang, Chao-Han Yang, Ji Wu, and Chao Zhang. 2024a. Can whisper perform speech-based in-context learning? In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 13421–13425. IEEE.

Siyin Wang, Chao-Han Huck Yang, Ji Wu, and Chao Zhang. 2024b. Can Whisper Perform Speech-Based In-Context Learning? In *Proc. ICASSP*.

A Appendix

A.1 Preprocessing

Some of the datasets include inconsistencies in formatting and script usage. For instance, certain utterances are fully marked with diacritics while others, sometimes from the same source, lack them entirely. To ensure consistency across all inputs, we apply a standard preprocessing pipeline inspired by Talafha et al. (2023). Specifically, we remove all punctuation except the % and @ symbols, strip diacritics, Hamzas, and Maddas, and convert Eastern Arabic numerals to their Western equivalents (e.g., ٢٩ to 29). Additionally, since our focus is not on code-switching, we exclude any Latin-script tokens from the data.

A.2 Model Settings

All experiments were conducted using the *transformers* and *datasets* libraries from HuggingFace. All audio segments were resampled to a sampling rate of 16kHz. Evaluations were performed on a single computing node equipped with 8 A10 GPUs (24GB each). For ASR systems, we employed: *Whisper:* `whisper-large-v3`[11] (1.55B parameters), *SeamlessM4T:* `seamless-m4t-v2-large`[12] (2.3B parameters), and *MMS:* `mms-1b-all`[13] (1B parameters).

For the retrieval-based components, we adopted the following extractors: **TF-IDF:** Character-level n-gram features using `analyzer="char_wb"` and `ngram_range=(3, 5)`. **Sentence Embeddings:** We used an off-the-shelf Arabic sentence encoder, [14], **Speech Embeddings:** Extracted from the final hidden states of the `whisper-large-v3` encoder. **Speaker Embeddings:** Derived from speaker verification with ECAPA-TDNN embeddings[15] trained on Voxceleb dataset (Desplanques et al., 2020). All models were used with their default hyperparameter settings unless otherwise specified.

A.3 Effect of Reversed Prompting on Hallucination and Output Fidelity

Table 2 presents manually selected examples illustrating the impact of reversed prompting on transcription quality. In each case, we compare the output of Whisper when conditioned on a standard SM4T-based prompt versus a reversed version of the same prompt. The examples highlight failure modes such as hallucinated phrases or overly short outputs in the standard prompt condition. Reversed prompting consistently recovers content that is more faithful to the reference transcription, with substantially lower WER.

A.4 Qualitative Analysis of Retrieval Modes

We manually analyzed 1,000 samples from the CV15 dev set to better understand the behavior of different retrieval extractors. Table 4 presents six representative query sentences along with the top matches returned by each method. TF-IDF consistently retrieved sentences with higher token-level overlap with the reference, resulting in more aligned surface-level matches. In contrast, dense text embeddings often returned semantically related but lexically divergent paraphrases, while speech and speaker embeddings frequently retrieved contextually unrelated content due to acoustic or speaker similarity. It is important to note that retrieval comparisons are based on the first-pass transcription, which serves as the input to the retrieval system. These qualitative observations align with our quantitative results, where TF-IDF achieved the lowest WER and CER on CV15 (17.89 / 7.96; n=1000; see Table 3). For example, when querying with the sentence

Reference	وايضا اعطى للعملية برمتها نوع من الـ
Whisper+prompt	مستشوفيات كثيرة في
Whisper+Rev	وايضا اعطى للعملية برمتها نوع من
WER (prompt)	1.00
WER (Rev)	0.14
Reference	ان الـ الخطاب السياسي مثلا اصبح مقيدا من طرف القضاء
Whisper+prompt	بل
Whisper+Rev	ان الخطاب السياسي مثلا اصبح مقيدا من طرف القضاء
WER (prompt)	1.00
WER (Rev)	0.10
Reference	عبر دستور ٢٠١١ وعبر العمل الحكومي بهذه الطريقة اي
Whisper+prompt	اشتركوا في القناة
Whisper+Rev	عبر دستور ٢٠١١ وعبر العمل الحكومي بهذه الطريقة
WER (prompt)	1.00
WER (Rev)	0.11

Table 2: Manually selected examples showing how reversed prompting mitigates hallucinations and improves WER.

Mode	WER/CER
Vanilla	22.84/9.65
TFIDF	**17.89/7.96**
Text Embedding	20.04/7.83
Speech	24.78/11.08
Speaker	27.16/13.26

Table 3: WER/CER using different feature extractors for text retrieval on CV15 (sample size = 1000).

[11] https://huggingface.co/openai/whisper-large-v3
[12] https://huggingface.co/facebook/seamless-m4t-v2-large
[13] https://huggingface.co/facebook/mms-1b-all
[14] https://huggingface.co/Omartificial-Intelligence-Space/Arabic-mpnet-base-all-nli-triplet
[15] https://huggingface.co/speechbrain/spkrec-ecapa-voxceleb

Reference	TF-IDF	Text Embedding	Speech Embeddings	Speaker Embeddings
(من الممكن انها لن تاتي غدا)	(من الممكن انها ستاتي)	(لم لن تكون هنا غدا)	(ربما من الافضل ان تاتي معنا)	(وداعيا الى الله باذنه وسراجا منيرا)
(لقد قابلته)	(قابلته يوما ما)	(قابلته يوما ما)	(اغنى خلقه بالمال)	(ساخذه)
(انك تكبر المشكلة)	(انا لست المشكلة)	(هو في مشكلة)	(وتشير باليد)	(واجتباه لنبوته)
(درس كل يوم لمدة ساعة ونصف)	(وجع ساعة ولا كل ساعة)	(لماذا تدرس كل يوم)	(نعم سيكفي الحليب حتى يوم الجمعة)	(عند سدرة المنتهى)
(ذهبت الى هناك ايضا)	(انا ايضا)	(اذهب الى هناك الان)	(يجب ان تذهب غربا)	(ترك لي توم رسالة)
(اتحسب سوء الظن يجرح في فكري)	(فاما سوء الظن بها فقد اختلف الناس فيه)	(كان سوء الظن بها يعمي عن محاسنها)	(انقذ الطفل بالمجازفة بحياته)	(عن ابي محمد الحسن بن علي بن ابي طالب)

Table 4: Examples of top retrieved sentences using different extractors. TF-IDF consistently preserves surface forms, while dense and acoustic features tend to retrieve semantically related but lexically or contextually divergent content. Sample size=1000

(من الممكن انها لن تاتي غدا), TF-IDF retrieves the closely related (من الممكن انها ستاتي), maintaining both structural and lexical overlap. In contrast, the text embedding method returns (لم لن تكون هنا غدا), semantically related but lexically distinct, while the speech-based method yields the more generic (ربما من الافضل ان تاتي معنا), and the speaker-based method retrieves (وداعيا الى الله باذنه وسراجا منيرا), which shares little contextual relevance. A similar pattern is seen for the query (انك تكبر المشكلة), where TF-IDF returns the precise phrase (انا لست المشكلة), while speech and speaker retrievals yield vague or acoustically aligned but semantically distant matches.

A.5 TTS Efficiency

We measure the performance of the TTS model by transcribing its synthetic output and comparing it with real speech under three conditions (i.e., MSA, Dialect, Accented MSA). See Figure 2.

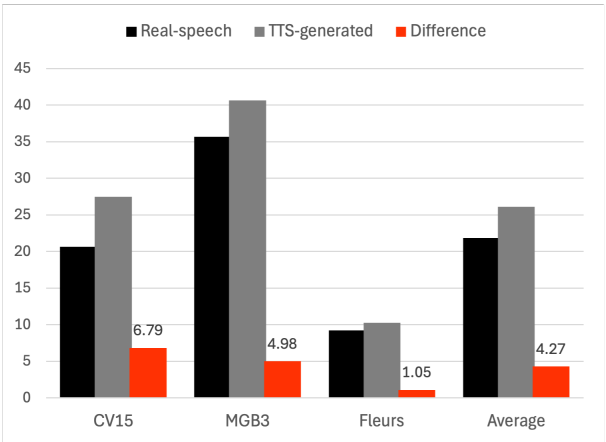


Figure 2: Comparison between the performance of Whisper on real speech vs. TTS-generated speech across different language settings (sample size=1000).