Octopus: Towards Building the Arabic Speech LLM Suite

Sara Althubaiti, Vasista Sai Lodagala, Tjad Clark, Yousseif Alshahawy, Daniel Izham, Abdullah Alrajeh, Aljawharah Bin Tamran, Ahmed Ali
HUMAIN, Riyadh, Saudi Arabia
{salthubaiti, vlodagala, TClark, yelshahwy, mizham, arajeh, ajbintamran, ahmed.ali}@humain.com

Abstract

We present **Octopus**, a first family of modular speech-language models designed for Arabic-English ASR, dialect identification, and speech translation. Built on Whisper-V3 and enhanced with large language models like ALLaM, LLaMA, and DeepSeek, Octopus bridges speech and text through a lightweight projection layer and Q-Former. To broaden its scope beyond speech, Octopus integrates BEATs, a general-purpose audio encoder allowing it to understand both linguistic and acoustic events. Despite its simplicity, this dual-encoder design supports robust performance across multilingual and code-switched scenarios. We also introduce **TinyOctopus**, a distilled variant using smaller models (Distil-Whisper + LLaMA3-1B / DeepSeek-1.5B), achieving competitive results with just a fraction of the parameters. Fine-tuning on synthetic code-switched data further boosts its performance. Octopus demonstrates the power of compact, extensible architectures in Arabic-centric speech modeling and sets the stage for unified multilingual audio-language understanding. The Octopus family models, along with the complete codebase, is publicly available[1].

1 Introduction and Related Work

The field of speech processing has witnessed remarkable advancements, particularly with the advent of large audio-language models (audio-LLMs). These models have shown promising capabilities in integrating acoustic information with natural language understanding, paving the way for more sophisticated human-AI speech interaction systems. Recent notable contributions in this area include **GAMA** (Ghosh et al., 2024), a general-purpose audio-LLM that integrates an LLM with various audio representations, demonstrating strong performance in audio understanding and complex reasoning tasks. Similarly, **Audio Flamingo** (Kong et al., 2024) proposes an audio language model designed for robust audio understanding, efficient few-shot learning, and multi-turn dialogue capabilities. Another significant effort, **AudioChatLlama** (Fathullah et al., 2024), explores extending LLMs to the speech domain, focusing on creating end-to-end systems that deliver consistent responses irrespective of speech or text inputs. Another relevant work, **ArTST** (Toyin et al., 2023), proposes an Arabic Text and Speech Transformer for ASR and speech translation. Similar to our approach, it supports Arabic-English tasks, but it follows a unified encoder-decoder transformer design trained end-to-end. In contrast, our Octopus framework integrates frozen high-capacity speech encoders (Whisper, BEATs) with frozen large language models via a modular Q-Former and projection layer, enabling flexible multitask extensions beyond ASR and translation. Furthermore, **Prompt-aware Mixture (PaM)** (Shan et al., 2025) has shown to improve Speech LLMs by utilizing multiple audio encoders, outperforming single-encoder models in various speech tasks.

However, a significant gap persists in their ability to perform fine-grained perception and complex reasoning in real-world, nuanced spoken language, especially for languages like Arabic, which present unique linguistic challenges such as rich morphology, dialectal variations, non-standard orthographic rules, and complex phonetics.

Recent efforts have aimed at developing more comprehensive evaluation benchmarks for large audio-language models to address these limitations. For instance, the MMSU benchmark (Wang et al., 2025) provides a massive multi-task spoken language understanding and reasoning framework, highlighting the need for models capable of fine-grained acoustic feature processing and linguistically-grounded reasoning. Addressing a specific gap in audio LLMs, **Audio Large Language Models Can Be Descriptive Speech Qual-**

[1] https://huggingface.co/ArabicSpeech/Octopus

Proceedings of The Third Arabic Natural Language Processing Conference, pages 425–435
November 8-9, 2025

ity Evaluators (Chen et al., 2025) presents a method for evaluating speech quality, enabling models to be more aware of the quality of the processed speech. Concurrently, **Towards Holistic Evaluation of Large Audio-Language Models: A Comprehensive Survey** (Yang et al., 2025) presents a systematic taxonomy for evaluating audio LLMs, categorizing evaluation benchmarks into four dimensions: *(i)* general auditory awareness and processing, *(ii)* knowledge and reasoning, *(iii)* dialogue-oriented ability, and *(iv)* fairness, safety, and trustworthiness, providing a structured overview of the fragmented landscape of audio LLM evaluations. These studies collectively underscore the ongoing challenges and the demand for robust and generalizable audio LLMS.

Through this work, we introduce Octopus, a novel family of multitask speech-LLMs specifically designed to address the some of the aforementioned challenges in Arabic speech understanding. We evaluate our models over multiple speech related tasks such as ASR (Bilingual and Code-switched), Speech-Translation (Arabic-to-English) and Arabic Dialect Identification (across 17 major dialects). Our analysis provides key insights about the size of LLMs to be used, the importance of multi-task and multi-lingual training.

2 Octopus LLM Family

The Octopus LLM family is a suite of Arabic-centric Speech Large Language Models (Speech-LLMs) developed for comprehensive understanding and generation from spoken Arabic across a wide range of dialects. Octopus is designed to perform several speech-language tasks, including automatic speech recognition (ASR), Arabic-to-English speech translation, and dialect identification, with strong performance across spontaneous and read speech.

Each model in the Octopus family combines a pre-trained audio encoder with a frozen large language model (LLM), connected through a lightweight trainable projection layer and an intermediate Q-Former for modality alignment. Extracting audio representations within the Octopus architecture is done using the Whisper encoder (Radford et al., 2023) (or its lightweight variant Distil-Whisper (Gandhi et al., 2023)) and BEATs encoder (Chen et al., 2022). While the Whisper encoder serves in extracting the semantic embeddings from the audios, the BEATs encoder provides the fine-grained acoustic representations. Given the general-purpose large scale training that these encoders have undergone, their parameters are not updated (frozen), while the Q-Former and projection layers are fine-tuned to bridge the audio and language modalities. It is to be noted that the high-level design of the Octopus suite of models has been inspired from the SALMONN architecture (Tang et al., 2023).

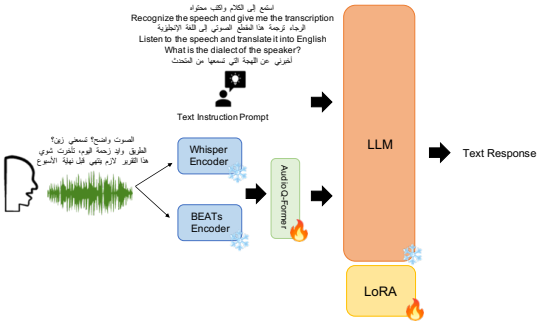

Figure 1: Overall architecture of the Octopus Speech-LLM family. Speech embeddings are extracted through frozen Whisper and BEATs encoders, aligned via a trainable Q-Former and projection layer, and then decoded by a frozen LLM.

Octopus supports a range of LLM backbones to accommodate various the training, deployment and downstream task requirements. These include lightweight models such as LLaMA 1B (Grattafiori et al., 2024) and DeepSeek 1.5B, as well as larger-scale options such as ALLaM-13B (Bari et al., 2024).

The models are trained on diverse Arabic speech corpora covering multiple dialects—including Saudi, Egyptian, Gulf, Levantine, and Mauritanian—spanning both ASR and dialect identification tasks. The Mauritanian dialect is not separately collected; it is part of the 17 dialects included in the publicly available ADI17 dataset used for dialect identification. For the translation component, we incorporate synthetic Arabic–English parallel corpora to enhance cross-lingual capabilities. A summary of all datasets, their availability, usage, and the train/dev splits (based on the official splits provided by the dataset creators when available) is presented in Table 1.

Figure 1 illustrates the overall architecture of the Octopus LLM family, including the dual-stream encoder design, Q-Former, projection layer, and language model integration.

2.1 Model Architecture

As illustrated in Figure 1, Octopus follows a modular encoder-decoder design that enables efficient alignment between audio inputs and a frozen large language model (LLM). The architecture is composed of four primary components: (1) audio encoders, (2) a cross-modal Q-Former, (3) a linear projection layer, and (4) an autoregressive decoder enhanced with LoRA-based adaptation. The finer architectural details are elaborated on in (Tang et al., 2023).

Training Strategy. During training, only the Q-Former, projection layer, and LoRA parameters are updated. Both audio encoders and the base language model remain frozen. This approach ensures efficient parameter usage, modularity, and robust generalization across multiple speech-language tasks, including ASR, Arabic-to-English translation, and dialect identification.

2.2 Datasets and Tasks

Our models are evaluated on three core tasks: automatic speech recognition (ASR), Arabic-to-English machine translation, and dialect identification. Table 1 summarizes the training data used for each task and the model configurations explored throughout our experiments.

To assess generalization and performance across a wide range of real-world scenarios, we evaluate our models on a diverse suite of test sets, selected to reflect variation in language, dialect, formality, and utterance length:

- **MGB2** (Ali et al., 2016) — 9.58 hours of broadcast news recordings for Arabic ASR, covering five dialects: Modern Standard Arabic (MSA), Gulf (GLF), Levantine (LEV), North African (NOR), and Egyptian (EGY).

- **LibriSpeech** — test-clean (5.40h) and test-other (5.34h) subsets for English ASR, representing clean and challenging noisy audio conditions.

- **TEDLIUM** (Hernandez et al., 2018) — test (2.62h) of English speech from TED talks, covering a wide range of topics, speakers, and accents. The dataset includes transcribed audio aligned at the word level and serves as a benchmark for ASR systems in lecture-style, spontaneous speech settings. .

- **ESCWA** — 2.77 hours of formal and semi-formal Arabic-English code-switched recordings from United Nations ESCWA meetings held in 2019, exhibiting intrasentential switching.

- **Mixat-All** (Ali and Aldarmaki, 2024) — 5.94 hours test set of Emirati-English speech sourced from two public podcasts featuring native Emirati speakers in both formal and conversational settings. From this, we extract 3.15 hours of pure code-switched segments and call it **Mixat-CS**.

- **In-house_long_files** — 25.33 hours of long-form Arabic ASR test set with 8–10 minute segments across five dialects (Saudi, MSA, Gulf, Jordanian, Egyptian), aimed at evaluating long-context and dialectal robustness.

These test sets enable robust evaluation across a spectrum of challenges, including multilinguality, code-switching, dialectal diversity, and long-form audio comprehension.

Machine Translation. For the Arabic-to-English translation task, we utilized transcribed speech segments from both our in-house dataset and the publicly available QASR corpus. To generate English translations, we employed GPT-4o, prompting it with standardized translation instructions. It is important to note that translation was conducted at the text level, not directly on the raw audio; the transcriptions served as the source for translation.

Upon manual and automatic review of the translated outputs, we observed a discrepancy between the number of segments used in the ASR task and those with valid translations—specifically, a reduction of approximately 43.25% for the in-house dataset and 1.17% for QASR. This discrepancy is primarily due to two factors: (1) GPT-4o occasionally failed to fully translate a segment, leaving residual Arabic phrases in the output, and (2) the model exhibited hallucination behavior in some instances, generating content unrelated to the source transcription.

Dialect Identification. For the dialect identification task, we utilized the ADI17 dataset (Shon et al., 2020), which was introduced as part of the VarDial Evaluation Campaign. The dataset comprises labeled speech segments from 17 Arabic dialects, with carefully curated training, development, and

test splits. It includes both audio and transcription metadata, supporting standardized evaluation protocols.

We follow the original split and setup described in ADI17 paper without modification. The dataset offers extensive dialectal coverage across North African, Levantine, and Gulf regions, making it well-defined for benchmarking Arabic dialect identification systems.

Automatic Speech Recognition (ASR). ASR evaluation was conducted across both in-house and public datasets, with transcriptions serving as ground truth. All audio was preprocessed to ensure consistent sampling rates and segment lengths. The datasets used for ASR include a broad spectrum of speaking styles, recording conditions, and dialectal diversity to ensure robust evaluation.

2.3 Multitask Learning Training

To enable generalization across speech-language tasks, we train our models using a multitask learning strategy that unifies automatic speech recognition (ASR), Arabic-to-English machine translation, and dialect identification within a single architecture. This framework allows the model to leverage shared acoustic-linguistic representations and instruction-tuned prompting.

Our training follows a progressive setup. We consistently begin by training on the ASR task using Arabic speech, as this data is readily available and provides a strong foundation for aligning audio and text. In subsequent experiments, we extend the training setup to a bilingual ASR configuration by incorporating English speech from LibriSpeech (clean and other) and TEDLIUM. This stage facilitates the model's exposure to multilingual speech patterns and supports robust cross-lingual audio-text alignment.

After establishing the ASR capabilities, we introduce supervision for the translation task using Arabic transcriptions paired with English translations, followed by the dialect identification task using dialect-labeled audio. This gradual inclusion of tasks enables better convergence and reduces task interference during training.

Each task is prompted using natural language instructions, with variations in both English and Arabic phrasing. This diversity in prompting enhances the model's instruction-following capabilities across languages and domains.

Training is performed in a multitask fashion, with task examples sampled in a round-robin manner across mini-batches. The total training loss is computed as a weighted sum of task-specific objectives:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{ASR}} \cdot \mathcal{L}_{\text{ASR}} + \lambda_{\text{MT}} \cdot \mathcal{L}_{\text{MT}} + \lambda_{\text{DID}} \cdot \mathcal{L}_{\text{DID}} \quad (1)$$

where λ values are hyperparameters that control the relative contribution of each task to the overall optimization. These weights are tuned empirically to mitigate task imbalance and prevent overfitting to high-resource tasks such as ASR.

Given the disparity in dataset sizes across tasks, we observed that naively optimizing all examples led to overfitting on ASR while underutilizing supervision from translation and dialect identification. To address this, we applied task sampling normalization by ensuring an equal number of updates per task within each epoch, regardless of the number of available examples. This effectively decouples task frequency from dataset size and forces the model to generalize across tasks.

We also explored tuning λ weights based on validation loss curves, which helped stabilize early convergence and preserved performance on low-resource tasks. Our findings are consistent with prior work (Tang et al., 2023) showing that careful balancing of task contributions is crucial for effective multitask training in speech-grounded LLMs.

This multitask strategy promotes parameter efficiency and improves generalization across tasks, particularly under dialectal variation, noisy transcriptions, and prompt phrasing diversity.

3 Experiments

To evaluate our proposed Octopus family, we conduct a series of experiments designed as research questions. Each question targets a specific aspect of our model's architecture, training setup, or generalization behavior. This format allows us to explore different task setups and component interactions, even when the results are not directly comparable under a single metric.

3.1 Q1: Does enriching the task and lingustic space improve overall performance?

We begin our exploration with a baseline model trained exclusively for Arabic ASR, denoted as **Ar_Octopus**, using Whisper-large-v3 as the encoder and ALLaM-13B as the frozen decoder. The training data includes only in-house Arabic ASR.

Table 1: Summary of the data splits used for each task, including total duration (in hours).

Dataset	# of Hours Train	Dev	Availability	Used in
ASR (Arabic)				
QASR	1,880.5	9.6	Public	TinyOctopus
In-house Arabic	13,392.1	142.7	Private	Octopus
ASR (English)				
LibriSpeech	960.0	10.5	Public	Octopus/TinyOctopus
TEDLIUM	453.8	1.6	Public	Octopus/TinyOctopus
ASR (Ar-En Code Switching)				
Synthetic (In-house TTS)	119.5	-	Private	TinyOctopus
Translation (Ar→En)				
Translated QASR (via GPT-4o)	1,858.4	9.6	Private	TinyOctopus
Translated in-house Arabic (via GPT-4o)	7,229.2	141.9	Private	Octopus
Dialect Identification				
ADI17	2,241.5	19.0	Public	TinyOctopus

To investigate the impact of task expansion, we progressively augment the task space. First, we build a **Bilingual_Octopus** model by introducing English ASR supervision from LibriSpeech (clean and other) and TED-LIUM corpora. Language-specific tokens (<ar>, <en>) are prefixed during training to each transcription to condition the model on the expected output language. This enables the decoder to distinguish between Arabic and English transcriptions, effectively guiding the shared encoder-decoder pathway in a multilingual context.

Next, we construct **Trans_Octopus** by introducing a translation task into the training loop. We use GPT-4o to translate the Arabic ASR transcripts (from both QASR and in-house) into English. These translated pairs are then treated as a parallel corpus for training. This step is inspired by recent work showing that auxiliary tasks can provide beneficial transfer signals in multimodal or multilingual setups (Zoph et al., 2016; Tang et al., 2020; Abdollahzadeh et al., 2021; Ma et al., 2024). In particular, multitask learning can regularize the model and improve representation sharing across tasks. All three models of Octopus shared the 15.1B number of parameters across different tasks, although 24M ones come from adapting LoRA with rank=8 and training the Q-former.

3.2 Q2: Can smaller distilled models match the performance of their larger counterparts?

Recent research has highlighted the potential of distilled models to retain much of the performance of their larger teacher models while significantly reducing computational and memory requirements. A notable example is Google's Distilling Step-by-Step (Hsieh et al., 2023), which demonstrates that smaller language models can outperform larger ones when trained with intermediate supervision and careful curriculum design, even with less data. Similarly, works such as DistilBERT (Sanh et al., 2019), TinyLLaMA (Zhang et al., 2024), and Distil-Whisper (Gandhi et al., 2023) have shown that distilled models, when fine-tuned for specific tasks, can match or exceed the performance of their full-sized counterparts on downstream benchmarks.

Motivated by these findings, we explore a *distilled audio-text pipeline* referred to as **TinyOctopus**. This setup replaces Whisper-large-v3 (1.5B parameters) with its distilled counterpart, Distil-Whisper-large-v3 (756M parameters), and replaces the decoder LLM with smaller variants, specifically LLaMA3–1B and DeepSeek-1.5B. The resulting speech-LLMs are TinyOctopus_LLAMA3-1B and TinyOctopus_Deepseek-1.5B respectively. These components are integrated into our TinyOctopus framework to investigate whether such downsizing can preserve or enhance performance in low-resource and multilingual scenarios.

For Arabic ASR, we train using the QASR dataset. For English ASR, we rely on standard high-resource benchmarks, namely LibriSpeech (both clean and other splits) and TEDLIUM. To enable cross-lingual supervision, we translate the QASR transcriptions to English using GPT-4o, providing data for the Arabic-to-English translation

Table 2: ASR Performance of Octopus variants across different task configurations. WER | CER, represent the word error rate and character error rate, respectively in percentage terms.

Dataset	Ar_Octopus	Bilingual_Octopus	Trans_Octopus	Whisper-large-v3	SeamlessM4T
Arabic ASR					
MGB2	16.5 | 6.5	15.2 | 6.8	**13.3 | 5.9**	16.2 | 7.9	17.2 | 8.4
English ASR					
test-clean	82.5 | 92.4	**2.6 | 1.4**	67.3 | 79.4	2.86 | 0.98	2.68 | 0.88
test-other	86.9 | 95.1	5.1 | 3.4	71.5 | 87.8	5.00 | 2.05	**5.07 | 1.94**
tedlium	101.9 | 77.4	**5.1 | 3.9**	85.2 | 63.6	11.92 | 4.44	86.51 | 62.22
Code-Switched (CS)					
Escwa	42.5 | 26.3	**40.8 | 27.1**	41.8 | 25.1	47.34 | 31.02	52.02 | 35.30
Mixat-ALL	22.0 | 9.0	**23.4 | 10.3**	24.3 | 10.6	29.08 | 15.07	32.83 | 16.88
Mixat-CS	26.4 | 12.4	28.5 | 14.9	**27.8 | 13.3**	34.83 | 20.57	38.23 | 21.84
Long-form					
In-house_long_files	25.4 | 13.0	24.9 | 12.5	**24.1 | 12.1**	26.7 | 15.2	29.3 | 18.6

task. Lastly, we introduce dialect identification as a task and train on the ADI17 dataset, which spans 17 Arabic dialects.

To further enhance performance towards code-switching, we conduct ASR-specific fine-tuning on augmented code-switched data. Specifically, we synthesize 119.50 hours of training audio from 99,999 code-switching utterances sourced from the `SA_TRAIN.txt` split provided by (Alharbi et al., 2024), which was generated using LLMs to expand Arabic-English code-switching text 1. We convert this synthetic text into speech using our internal in-house TTS system.

Our findings as elaborated in section 4.1.1 suggest that distilled and compact models, when supported by high-quality synthetic data and targeted fine-tuning, can rival or even surpass larger counterparts in multilingual and multitask audio understanding—especially in code-switched or low-resource conditions.

Furthermore, TinyOctopus leverages the compact Distil-Whisper encoder (756M parameters) alongside smaller LLMs. Specifically, the variant with LLaMA3-1B totals approximately 1.75B parameters, while the version with DeepSeek-1.5B version has about 2.25B parameters. The parameter-efficient fine-tuning conducted using LoRA (rank=8), requires only $\sim$12M and $\sim$13M parameters to be trained in each setup, respectively. This allows us to retain strong performance with minimal computational cost.

4 Results, Analysis and Discussion

This section presents the performance of the proposed models across automatic speech recognition (ASR), speech translation, and dialect identification tasks.

4.1 ASR Beyond the Basics: How Far Can Multitask and Distilled Models Stretch

Tables 2 and 3 demonstrate the results of the various models from the Octopus suite on monolingual, code-switched, and long-form ASR test sets which have been described in section 2.2.

Table 2 shows the ASR performance of Octopus variants alongside recent strong baselines, Whisper-large-v3 and SeamlessM4T. As expected, Ar_Octopus performs quite well on MGB2, while under-performing on test-clean, test-other and tedlium. Introducing an additional language with language-specific tokens, as done in the case of Bilingual_Octopus, results in improved performance on MGB2, while showing impressive error rates on the English testsets. Although introducing additional speech data in terms of a new language (English) helped the model generalize better, a modeling choice, such as the use of language-specific tokens certainly helped the model distinguish between the acoustics of the two languages and associating them with the corresponding transcriptions. Introducing an additional, yet allied task such as speech-translation in the case of Trans_Octopus improves the error rates on MGB significantly, thereby validating the effectiveness of the multi-task training strategy. It is interesting to note that the error rate on English test sets has also reduced significantly compared to the Ar_Octopus model, though the model has not been trained on any English ASR data. This is most likely the case because of the shared output space of tokens between English speech recognition and Ar->En speech translation. Our approach of multi-task training resulted in a 19.4% relative WER improvement for the Trans_Octopus model over the Ar_Octopus model on Arabic. Bilingual

Table 3: ASR Performance of the TinyOctopus variants and their fine-tuned versions. WER | CER represent the word error rate and character error rate, respectively in percentage terms.

Dataset	TinyOctopus_LLaMA3-1B	TinyOctopus_LLaMA3-1B_finetuned	TinyOctopus_DeepSeek-1.5B	TinyOctopus_DeepSeek-1.5B_finetuned
Arabic ASR				
MGB2	22.6 | 15.7	16.1 | 9.5	23.2 | 15.8	**15.5 | 9.2**
English ASR				
test-clean	7.5 | 5.7	**3.1 | 1.3**	7.7 | 5.8	7.6 | 5.7
test-other	11.3 | 8.0	6.9 | 3.5	11.5 | 8.2	11.3 | 8.0
Code-Switched (CS)				
Escwa	42.5 | 26.9	**40.3 | 24.4**	43.6 | 27.8	41.8 | 26.3
Mixat-All	35.2 | 19.6	**34.1 | 19.3**	37.1 | 21.1	35.5 | 19.9
Mixat-CS	40.2 | 24.2	**36.2 | 21.4**	41.2 | 25.2	39.9 | 24.2
Long-form				
In-house_long_files	44.3 | 29.1	**42.8 | 26.9**	47.0 | 32.7	43.7 | 31.5

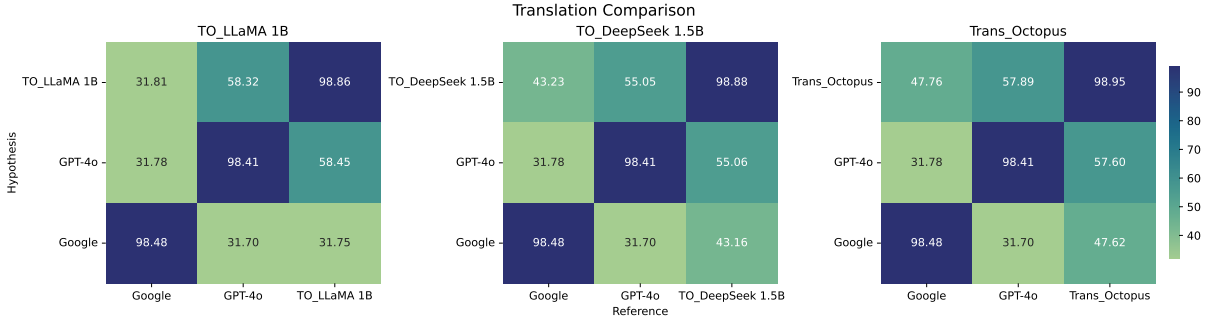

Figure 2: Pair-wise BLEU score comparison between Google, TinyOctopus (TO) / Trans_Octopus and GPT-4o

Model/System	CoVoST2 (Ar→En) BLEU ↑	CoVoST2 (Ar→En) BERT-F1 ↑	FLEURS (Ar→En) BLEU ↑	FLEURS (Ar→En) BERT-F1 ↑
Whisper-large-v3	28.8	0.53	15.1	0.47
SeamlessM4T	33.7	0.55	**23.9**	0.56
Trans-Octopus	**38.6**	**0.64**	23.2	**0.58**
TO-Llama-1B	33.9	0.61	20.5	0.53
TO-DeepSeek-1.5B	33.6	0.61	20.8	0.53

Table 4: Translation performance on CoVoST2 and FLEURS (Arabic→English) using BLEU (lexical) and BERTScore F1 (semantic).

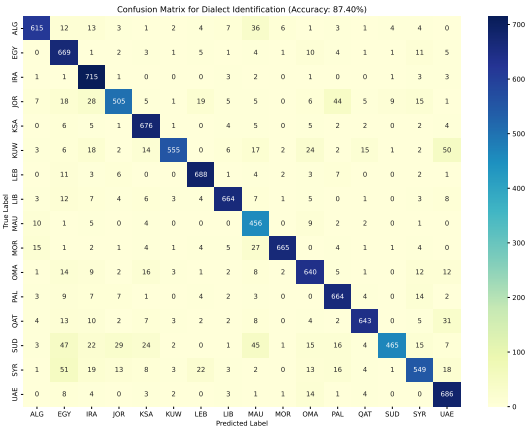

Figure 3: Confusion matrix for dialect identification on the QASR test set by the TinyOctopus_LLAMA3-1B model, showing true vs. predicted labels for 17 Arabic dialects.

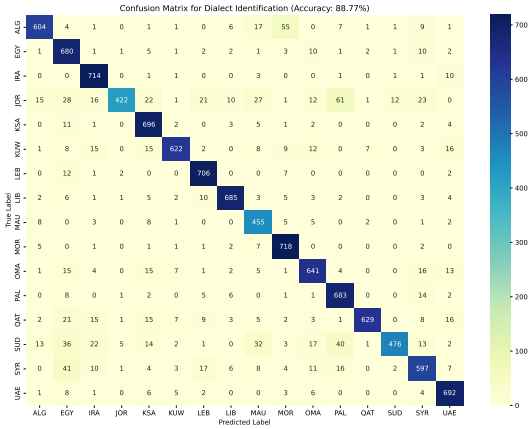

Figure 4: Confusion matrix for dialect identification on the QASR test set by the TinyOctopus_DeepSeek-1.5B model, showing true vs. predicted labels for 17 Arabic dialects.

training with language specific tokens resulted in a 86.2% average absolute WER improvement over the English testsets for the Bilingual_Octopus over the Ar_Octopus model. These results support our hypothesis that incorporating complementary tasks, particularly those that share encoder-level features or decoder-level objectives can significantly enhance learning and improve the downstream performance.

Coming to the performance of the Tiny Octopus models in table 3, we notice that the error rates

are higher compared to the task-specific models in Table 2. This is to be attributed, primarily to the considerable reduction in the Arabic ASR training data for the Tiny Octopus models. As the TinyOctopus models have been trained on 3 tasks (Bilingual ASR, Speech-Translation and dialect identification), the multilingual and multi-task training approach helps the models achieve moderate word error rates over the monolingual test sets. The Octopus models on the other hand have the handicap of being trained on fewer number of tasks or being monolingual in terms of the speech data. Fine-tuning the TinyOctopus models on (Ar-En) code-switching data does improve the error rates across languages significantly, thereby overcoming the handicap of having a smaller decoder (LLM) compared to the Octopus models. This shows that fine-tuning and multi-task training are far more effective compared to having larger LLMs as decoders on limited number of tasks.

4.1.1 Code-Switching ASR

The trend of introducing additional languages and allied tasks to the model training results in improved performance on code-switching ASR and this can be noticed in Table 2. The error rates on code-switching test sets improves as we move across, from Ar_Octopus to Bilingual_Octopus and Trans_Octopus.

The Tiny Octopus models greatly benefit from fine-tuning on code-switching data resulting in significant reduction of error rates for the TinyOctopus_LLAMA3-1B_finetuned and TinyOctopus_DeepSeek-1.5B_finetuned compared to their pre-trained counterparts. Given that the code-switching data is Ar-En, fine-tuning helps in improving the WERs on the code-switching test sets, while also achieving significant improvements over the monolingual test sets. The fine-tuning approach avoids any catastrophic forgetting on the monolingual tasks because, the speech encoder and the LLM parameters are frozen and only the parameters of the Q-former and the adapter layers are updated.

4.1.2 Long-form Speech Recognition

The long-form benchmark, with audio files averaging 8–10 minutes and representing mixed dialects, challenges the generalization capabilities of models trained on more concise and dialect-specific data. As the voice-activity detection (VAD) module has been observed to be mediocre in terms of its accuracy, we use an external Voice-Activity detection (VAD) model such as Silero-VAD (Team, 2021) to segment the speech over this benchmark.

Adhering to the trend on monlingual Arabic, Trans_Octopus outperforms Bilingual_Octopus which in turn outperforms Ar_Octopus on long-form ASR (as shown in Table 2, thereby reinforcing the importance of multitask and multilingual training.

The huge increase in error rates for the TinyOctopus models in Table 3 compared to the models in Table 2 is expected, largely due to the amount of Arabic training data the models have been exposed to. The Tiny-Octopus models have been exposed to just $\sim 1,900$ hours of Arabic data coming from QASR, whereas the Octopus models have a volume and dialectal depth for having been trained on $\sim 13,400$ hours of in-house Arabic speech.

To further investigate this gap, we conducted a small-scale experiment by augmenting the QASR training set with our in-house Arabic dataset, and retraining the best TinyOctopus variant (`TinyOctopus_LLaMA3-1B`). The resulting performance improved substantially, achieving a WER | CER of **24.9 | 13.1**, compared to the previous **44.3 ; 29.1**. This highlights the importance of both training volume and dialectal coverage for long-form ASR, especially when using compact and distilled architectures.

Fine-tuning the TinyOctopus models improves the performance too (as shown in Table 3). However, the gains obtained from scaling up and dialectal coverage of data, still outweigh the gains from fine-tuning.

4.2 Can Multi-task Models Match GPT-4o and Google in Dialectal Translation?

As the Trans_Octopus, TinyOctopus_LLAMA3-1B and TinyOctopus_DeepSeek-1.5B have one of their training objectives as speech translation, in this subsection, we discuss their efficacy over the same. We evaluate the translation capabilities of the models over the test set of QASR (Mubarak et al., 2021).

As described in sections 3.1 and 3.2, the translation references for training have been synthesized using GPT-4o, which has been tasked with translating the ASR transcripts. The lack of real speech translation data across Arabic dialects has resulted in taking such a route. Now, in order to evaluate the speech translation capabilities of our models, we do so by comparing their results against the machine

translation capabilities of Google[2] and GPT-4o systems. It is to be noted that speech-translation as a task is much more complex and hard compared to machine translation. This is because, speech translation deals with two modalities (speech and text), while machine translation is a task over the same modality (text). In addition, unlike ASR, speech translation is not monotonic in relation between the input and its output.

In spite of these limitations, from Fig. 2 we notice that the Octopus and TinyOctopus models have consistenly outperfomed Google and GPT-4o's translation capabilities from Arabic-to-English when compared against each other. Fig. 2 provides a pair-wise comparison of models by considering the reference and hypothesis from each of the models and comparing against the others. Considering the volume of the Arabic speech and the scale of the model, **Trans_Octopus** emerges as the best speech-translation model (Ar->En) within the Octopus family.

In addition to the dialectal QASR evaluation, we further benchmarked our models on established human-annotated datasets, CoVoST2 (Wang et al., 2020) and FLEURS (Conneau et al., 2022) , to situate our results within the broader speech translation literature. Table 4 reports BLEU (lexical) and BERTScore F1 (semantic). We observe that **Trans_Octopus** achieves the best performance on both datasets, with BLEU scores of 38.6 on CoVoST2 and 23.2 on FLEURS, coupled with the highest semantic fidelity (BERT-F1 = 0.64 and 0.58, respectively). The TinyOctopus variants (TO-LLaMA3-1B and TO-DeepSeek-1.5B) also perform competitively, outperforming Whisper-large-v3 and SeamlessM4T in both lexical and semantic quality. These results reinforce our central claim, multi-task training in the Octopus family not only enables strong dialectal performance but also generalizes well to established public benchmarks. Trans_Octopus emerges as the most capable Ar→En speech translation model across both in-house and public evaluations.

4.3 Can One Model Understand 17 Dialects?

Upon evaluating our TinyOctopus models TinyOctopus_LLAMA3-1B and TinyOctopus_DeepSeek-1.5B on the test set of ADI-17 (Shon et al., 2020), we notice that both of these models achieve impressive accuracies in identifying the 17 Arabic dialects. While the TinyOctopus_LLAMA3-1B model achieves 87.4% accuracy over the benchmark, the TinyOctopus_DeepSeek-1.5B model outperforms it at 88.7% accuracy. Figures 3 and 4 illustrate the dialect-wise identification performance of these models.

[2]https://github.com/nidhaloff/deep-translator

5 Conclusion and Future Work

In this paper, we introduced **Octopus**, a first-of-its-kind Arabic Speech-LLM suite designed to address the rich diversity of Arabic dialects and their interaction with English. Through extensive experiments, we evaluated key architectural and training choices across Arabic/English ASR, code-switching recognition, dialect identification, and Arabic–English translation. Recent Speech-LLMs such as GAMA, AudioFlamingo-3, Canary, and Qwen2.5-Audio show strong multilingual progress, yet their performance on dialectal Arabic remains limited. Even high-capacity general-purpose models often failed to produce accurate dialectal translations highlighting the gap that Octopus fills. It is also important to note that, **Octopus is not designed as a zero-shot system** but follows a supervised multi-task paradigm, where tasks are explicitly taught using curated datasets. While zero-shot transfer is an interesting future direction, it is beyond the present scope. By explicitly targeting Arabic and code-switched speech, Octopus establishes a modular framework for under-resourced languages. Future work will expand to additional tasks (e.g., speaker recognition, emotion detection) and introduce an **Arabic Speech Understanding Leaderboard** to benchmark progress across dialects, tasks, and models.

References

Milad Abdollahzadeh, Touba Malekzadeh, and Ngai-Man Man Cheung. 2021. Revisit multimodal meta-learning through the lens of multi-task learning. *Advances in Neural Information Processing Systems*, 34:14632–14644.

Sadeen Alharbi, Reem Binmuqbil, Ahmed Ali, Raghad Aloraini, Saiful Bari, Areeb Alowisheq, and Yaser Alonaizan. 2024. Leveraging llm for augmenting textual data in code-switching asr: Arabic as an example. *Proceedings of SynData4GenAI*.

Ahmed Ali, Peter Bell, James Glass, Yacine Messaoui, Hamdy Mubarak, Steve Renals, and Yifan Zhang. 2016. The mgb-2 challenge: Arabic multi-dialect broadcast media recognition. In *2016 IEEE Spoken*

Language Technology Workshop (SLT), pages 279–284. IEEE.

Maryam Al Ali and Hanan Aldarmaki. 2024. Mixat: A data set of bilingual emirati-english speech. *arXiv preprint arXiv:2405.02578*.

M Saiful Bari, Yazeed Alnumay, Norah A Alzahrani, Nouf M Alotaibi, Hisham A Alyahya, Sultan AlRashed, Faisal A Mirza, Shaykhah Z Alsubaie, Hassan A Alahmed, Ghadah Alabduljabbar, and 1 others. 2024. Allam: Large language models for arabic and english. *arXiv preprint arXiv:2407.15390*.

Chen Chen, Yuchen Hu, Siyin Wang, Helin Wang, Zhehuai Chen, Chao Zhang, Chao-Han Huck Yang, and Eng Siong Chng. 2025. Audio large language models can be descriptive speech quality evaluators. *arXiv preprint arXiv:2501.17126*.

Sanyuan Chen, Yu Wu, Chengyi Wang, Shujie Liu, Daniel Tompkins, Zhuo Chen, and Furu Wei. 2022. Beats: Audio pre-training with acoustic tokenizers. *arXiv preprint arXiv:2212.09058*.

Alexis Conneau, Min Ma, Simran Khanuja, Yu Zhang, Vera Axelrod, Siddharth Dalmia, Jason Riesa, Clara Rivera, and Ankur Bapna. 2022. Fleurs: Few-shot learning evaluation of universal representations of speech. *arXiv preprint arXiv:2205.12446*.

Yassir Fathullah, Chunyang Wu, Egor Lakomkin, Ke Li, Junteng Jia, Yuan Shangguan, Jay Mahadeokar, Ozlem Kalinli, Christian Fuegen, and Mike Seltzer. 2024. Audiochatllama: Towards general-purpose speech abilities for llms. *NAACL*.

Sanchit Gandhi, Patrick von Platen, and Alexander M Rush. 2023. Distil-whisper: Robust knowledge distillation via large-scale pseudo labelling. *arXiv preprint arXiv:2311.00430*.

Sreyan Ghosh, Sonal Kumar, Ashish Seth, Chandra Kiran Reddy Evuru, Utkarsh Tyagi, S Sakshi, Oriol Nieto, and Ramani Duraiswami. 2024. GAMA: A large audio-language model with advanced audio understanding and complex reasoning abilities. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 6288–6313.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

François Hernandez, Vincent Nguyen, Sahar Ghannay, Natalia Tomashenko, and Yannick Esteve. 2018. Tedlium 3: Twice as much data and corpus repartition for experiments on speaker adaptation. In *Speech and Computer: 20th International Conference, SPECOM 2018, Leipzig, Germany, September 18–22, 2018, Proceedings 20*, pages 198–208. Springer.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alexander Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. 2023. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. *arXiv preprint arXiv:2305.02301*.

Zhifeng Kong, Arushi Goel, Rohan Badlani, Wei Ping, Rafael Valle, and Bryan Catanzaro. 2024. Audio flamingo: A novel audio language model with few-shot learning and dialogue abilities. *Proceedings of the 41st International Conference on Machine Learning (ICML)*.

Rao Ma, Mengjie Qian, Yassir Fathullah, Siyuan Tang, Mark Gales, and Kate Knill. 2024. Cross-lingual transfer learning for speech translation. *arXiv preprint arXiv:2407.01130*.

Hamdy Mubarak, Amir Hussein, Shammur Absar Chowdhury, and Ahmed Ali. 2021. Qasr: Qcri aljazeera speech resource–a large scale annotated arabic speech corpus. *arXiv preprint arXiv:2106.13000*.

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

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Weiqiao Shan, Yuang Li, Yuhao Zhang, Yingfeng Luo, Chen Xu, Xiaofeng Zhao, Long Meng, Yunfei Lu, Min Zhang, Hao Yang, Tong Xiao, and Jingbo Zhu. 2025. Enhancing speech large language models with prompt-aware mixture of audio encoders. *arXiv preprint arXiv:2502.10098*.

Suwon Shon, Ahmed Ali, Younes Samih, Hamdy Mubarak, and James Glass. 2020. Adi17: A fine-grained arabic dialect identification dataset. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8244–8248. IEEE.

Changli Tang, Wenyi Yu, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun Ma, and Chao Zhang. 2023. Salmonn: Towards generic hearing abilities for large language models. *arXiv preprint arXiv:2310.13289*.

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2020. Multilingual translation with extensible multilingual pretraining and finetuning. *arXiv preprint arXiv:2008.00401*.

Silero Team. 2021. Silero vad: pre-trained enterprise-grade voice activity detector (vad), number detector and language classifier. https://github.com/snakers4/silero-vad.

Hawau Olamide Toyin, Amirbek Djanibekov, Ajinkya Kulkarni, and Hanan Aldarmaki. 2023. Artst: Arabic text and speech transformer. *arXiv preprint arXiv:2310.16621*.

Changhan Wang, Anne Wu, and Juan Pino. 2020. Covost 2: A massively multilingual speech-to-text translation corpus. *Preprint*, arXiv:2007.10310.

Dingdong Wang, Jincenzi Wu, Junan Li, Dongchao Yang, Xueyuan Chen, Tianhua Zhang, and Helen Meng. 2025. MMSU: A massive multi-task spoken language understanding and reasoning benchmark. *arXiv preprint arXiv:2506.04779*.

Chih-Kai Yang, Neo S Ho, and Hung-yi Lee. 2025. Towards holistic evaluation of large audio-language models: A comprehensive survey. *arXiv preprint arXiv:2505.15957*.

Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. 2024. Tinyllama: An open-source small language model. *arXiv preprint arXiv:2401.02385*.

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer learning for low-resource neural machine translation. *arXiv preprint arXiv:1604.02201*.