UniverSLU: Universal Spoken Language Understanding for Diverse Tasks with Natural Language Instructions

Anonymous ACL submission

Abstract

Recent studies leverage large language models with multi-tasking capabilities, using natural language prompts to guide the model's behavior and surpassing performance of task-specific models. Motivated by this, we ask: *can we build a single model that jointly performs various spoken language understanding (SLU) tasks?* We start by adapting a pre-trained automatic speech recognition model to multi-task using single-token task specifiers. We enhance this approach through *instruction tuning*, i.e., finetuning by describing the task using natural language instructions followed by the list of label options. Our approach can generalize to new task descriptions for the seen tasks during inference, thereby enhancing its user-friendliness. We demonstrate the efficacy of our *single* multi-task learning model "UniverSLU" for 12 speech classification and sequence generation task types spanning 17 datasets and 9 languages. On most tasks, UniverSLU achieves competitive performance and often even surpasses task-specific models. Additionally, we assess the *zero-shot* capabilities, finding that the model generalizes to new datasets and languages for seen task types.

1 Introduction

Spoken language understanding (SLU) refers to the task of inferring semantic meaning or linguistic structure from spoken utterances. Similar to prior work (Chang et al., 2023), we use the term broadly to refer to understanding content (Warden, 2018), speaker characterstics (Hechmi et al., 2021), paralinguistics (Busso et al., 2008) and semantic (Bastianelli et al., 2020) directly from speech.[1] In this work, we define a task as a specific combination of *task type* and *dataset*. A *task type* refers to the type of SLU operation being carried out, like intent classification (IC) or emotion recognition (ER). For instance, performing the same task type (IC) on SNIPS (Saade et al., 2018) and FSC (Lugosch et al., 2019) datasets are different tasks.

SLU tasks are traditionally approached with "task-specific" models, i.e. having a separate model for each task. On the other hand, the use of multi-task learning (MTL) approaches in natural language processing (NLP) (Chen et al., 2021; Zhang and Yang, 2017) and speech processing (Zhang et al., 2023; Radford et al., 2022) has garnered significant interest. They offer cost-efficiency by eliminating the need for separate models for each task. MTL has also been employed for SLU tasks (Arora et al., 2022a,b; Huang et al., 2022), primarily aiming to improve the performance of tasks with limited labeled data using an auxiliary automatic speech recognition (ASR) objective.

The rise of large language models (LLMs) (Radford et al., 2019) has sparked interest in MTL, where the selection of appropriate "prompts" (Liu et al., 2023) enables manipulation of the model's behavior to perform specific tasks without additional task-specific parameters. LLMs fine-tuned with natural language instructions (Wei et al., 2022) describing the task show substantial improvement in zero-shot performance. Building upon LLM's success in solving various NLP tasks, there is growing interest in leveraging the "pre-train, prompt, and predict" (Liu et al., 2023) paradigm.

Efforts have been made to explore similar prompt-based paradigms for speech processing tasks (Gao et al., 2022; Peng et al., 2023a). SpeechPrompt (Chang et al., 2022, 2023) employs trainable task-specific prompt vectors with generative spoken language models (GSLMs) (Lakhotia et al., 2021). However, this model still lags behind task-specific baselines in most speech classification tasks and particularly struggles on sequence generation tasks. Recently, there has been work on instruction tuning for speech processing tasks (Lai et al., 2023). However, these approaches mainly

[1] Additionally, we also include audio understanding tasks. Table 7 in Appendix briefly outlines the diversity in SLU tasks.

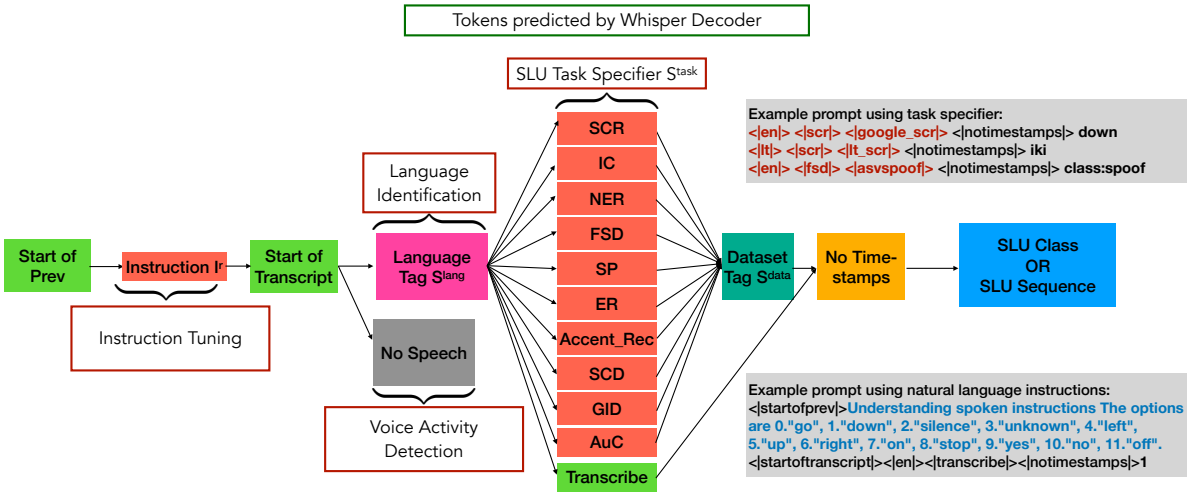

Figure 1: Schematics of our discrete prompt-based MTL formulation. Our architecture comprises an encoder-decoder architecture pre-trained using OpenAI's Whisper model, as detailed in Section 4. The figure illustrates the sequence of tokens to be predicted by the decoder in our MTL training format. We explore the use of single-token task specifiers or natural language phrases to describe the task, as discussed in Sections 4.1 and 4.2, respectively.

concentrate on ASR and related tasks and are often not able to perform diverse SLU tasks.

Motivated by the potential of prompt-based MTL, there has been interest in building "universal" ASR models (Zhang et al., 2023; Li et al., 2021) that can perform ASR across many languages. We extend this work by building a "universal" SLU model that can match or even outperform state-of-the-art (SOTA) task-specific SLU models. We leverage the Whisper (Radford et al., 2022) model, which has been pre-trained on labeled data for ASR and speech translation (ST) tasks using only 2 task specifiers, namely ⟨|transcribe|⟩ and ⟨|translate|⟩ respectively. We first generalize the Whisper model to a wide range of SLU tasks by utilizing single token *task specifiers*, as shown in Fig. 1, drawing inspiration from Whisper's pre-training formulation. We show that unlike the most comparable prior multi-task approach, SpeechPrompt v2 (Chang et al., 2023), our MTL model can outperform or achieve competitive performance to task-specific baselines across many classification and sequence generation tasks. However, the use of single-token task specifiers not only constrains our model's ability to handle variations in input format, affecting its user-friendliness, but also prevents it from performing unseen tasks in a zero-shot manner.

Hence, we experiment with utilizing natural language instructions (Wei et al., 2022), such as "**Understanding spoken instructions**" that describe each task as a prompt (shown as "example prompt using natural language instruction" in Fig. 1). For classification tasks, we *include a list of options* within the instruction. Our approach generalizes to new paraphrases describing the same task, enhancing its user-friendliness. We assess our model's capability of handling unseen tasks, covering new datasets and languages for the known task types, as well as on fully unseen task types. This is achieved by conducting inference using a task type description accompanied by the list of options specific to the given dataset as the instruction. Our preliminary investigation reveals that our model surpasses random and majority baselines in zero-shot inference on unseen datasets and languages, although it does not match supervised performance.

Our experiments encompass 10 speech classification and 2 sequence generation task types, covering 17 *publicly available* datasets based on 9 languages. Our proposed approach using natural language instruction significantly advances the SOTA on 10 classification tasks and achieves competitive performance to task-specific baselines on sequence generation tasks. We will make our source code and models publicly available for reproducibility. The key contributions of our work are:

- We introduce a novel approach that leverages human-interpretable instructions combined with a list of option labels as prompts for SLU tasks. We demonstrate that our method can generalize to unseen paraphrases.
- Our proposed approach can generalize effectively to new datasets and languages for seen task types in a zero-shot manner.
- We build a *single* MTL model across 17 SLU datasets that performs competitively or outperforms task-specific models on most tasks.

2 Related Work

Multi-task Learning (MTL) (Chen et al., 2021; Caruana, 1997) concurrently trains a model to perform multiple tasks, allowing the model to learn generalizable features instead of task-specific ones. MTL has been extensively explored in various NLP settings (Wu et al., 2019; Zhao et al., 2019; Zhuang and Liu, 2019; Pfeiffer et al., 2020), ranging from similar tasks (de Souza et al., 2015) to the same task type across multiple domains and languages (Yang and Hospedales, 2015). MTL can surpass task-specific models in certain scenarios, especially for tasks with limited labeled data (Chan et al., 2021). Similar investigations have been conducted in speech processing, such as adding auxiliary tasks in ASR (Prabhavalkar et al., 2023; Kim et al., 2017; Pironkov et al., 2016; Toshniwal et al., 2017) and for other tasks (Anastasopoulos and Chiang, 2018; Zheng et al., 2022; Cai et al., 2021; Sigtia et al., 2020), primarily using an auxiliary ASR objective. However, these approaches have covered very few SLU benchmarks, and we extend this line of research by MTL across 17 SLU datasets.

LLMs for speech processing: Recently, there has been significant interest in enhancing LLMs with speech processing and understanding capabilities (Gong et al., 2023b; Rubenstein et al., 2023; Fathullah et al., 2023a; Maiti et al., 2023). These approaches typically involve multi-tasking on a wide range of speech processing tasks, demonstrating robust performance across multiple benchmarks. LauraGPT (Wang et al., 2023) is one such approach that uses Whisper-style task specifiers with Qwen (Bai et al., 2023) to handle various speech processing and generation tasks. Some works have explored preserving LLM capabilities, such as understanding "instructions" to perform tasks in a zero-shot manner (Gong et al., 2023a; Tang et al., 2023; Chu et al., 2023; Fathullah et al., 2023b). LTU-AS (Gong et al., 2023a), concatenates text instruction with Whisper encoder output fed into Vicuna LLM (Chiang et al., 2023) to perform a range of speech and audio tasks. SALMONN (Tang et al., 2023), employs both speech and audio representation into Vicuna LLM for "instruction tuning" across various tasks.

However, while these efforts primarily concentrate on ASR (Yu et al., 2023; Lakomkin et al., 2023; Fathullah et al., 2023a) and ST (Chen et al., 2023), our approach focuses on a diverse array of SLU tasks (see section 6.2 and A.4). Furthermore, our novel approach, involving the addition of the list of options in the instruction, not only improves performance but also enhances model generalization to new datasets and languages for the trained task types. Our results in section 6.2 demonstrate our ability to outperform LLM-based approaches on the subset of SLU tasks they are trained on. Moreover, we distinguish ourselves as the first to exhibit *instruction tuning capabilities with a decoder of speech foundation model* to the best of our knowledge, that has the important advantage of using much fewer parameters and lower computation.

3 Problem Formulation

Our objective is to build a prompt-based MTL framework capable of performing various speech classification and sequence generation tasks. Our model takes as input a sequence of speech features, denoted X, as input for all downstream tasks. For a specific downstream task r, our aim is to predict the corresponding label sequence $Y^r = \{y^r_o \in \mathcal{L}^r | o = 1, \ldots, O^r\}$, where $\mathcal{L}^r$ represents the label set and O^r is the length of the output sequence for that particular task. Table 7 in the Appendix provides a comprehensive description of the diverse SLU tasks covered in this work, including example label sequences Y^r for each task. For classification tasks, $O^r = 1$. To achieve our aim, the prompting paradigm introduces a task-specific prompt S^r. Using maximum a posteriori (MAP) decision theory, the MTL model estimates $\hat{Y}^r$ as:

$$\hat{Y}^r = \text{argmax}\, P(Y^r | X, S^r) \tag{1}$$

Prior studies like SpeechPrompt (Chang et al., 2022, 2023) have modelled the prompting formulation using GSLMs. They prepend the embedding (embedding layer $\text{Emb}(\cdot)$) of discrete tokens U with continuous task-specific prompt vectors P^r. These vectors are learned for each task by maximizing the posterior distribution $P(Y^r | \text{Emb}(U), P^r)$.

In our approach, instead of relying on pre-trained model solely for generating discrete tokens for GSLM, we finetune the entire pre-trained model. Furthermore, we use discrete prompts corresponding to a sequence of tokens. Starting from a natural extension of pre-training formulation in speech foundation models, we first investigate using single token "task specifiers" as prompts. During pre-training, Whisper (Radford et al., 2022) utilizes only two task specifiers, namely $\langle|\text{transcribe}|\rangle$ and $\langle|\text{translate}|\rangle$, to facilitate MTL across ASR and ST

S^{task}	S^{data}	S^{lang}	**Example Label**	**Generated Prompt**	
				Task Specifier	Natural Language Instruction (I^r)
SCR	Google SC v1	En	down	⟨|**en**|⟩ ⟨|**scr**|⟩ ⟨|**google_scr**|⟩ ⟨|notimestamps|⟩ **down**	**Understanding spoken instructions. The options are 0."down", ⋯** ⟨|startoftranscript|⟩ ⟨|en|⟩ ⟨|transcribe|⟩ ⟨|notimestamps|⟩ **0**
	Lithuanian SC	Lt	iki	⟨|**lt**|⟩ ⟨|**scr**|⟩ ⟨|**lt_scr**|⟩ ⟨|notimestamps|⟩ **iki**	**Understanding spoken instructions. The options are 0."iki", ⋯** ⟨|startoftranscript|⟩ ⟨|lt|⟩ ⟨|transcribe|⟩ ⟨|notimestamps|⟩ **0**
LID	Voxforge	En, Es, Fr, De, Ru, It	Ru	⟨|**ru**|⟩	**Determining the language in spoken speech. The options are 0."⟨|ru|⟩", ⋯** ⟨|startoftranscript|⟩ ⟨|**ru**|⟩
FSD	ASVspoof	En	spoof	⟨|**en**|⟩ ⟨|**fsd**|⟩ ⟨|**asvspoof**|⟩ ⟨|notimestamps|⟩ **class:spoof**	**Tell apart speech synthesis and voice conversion from authentic speech. The options are 0."bonafide", 1."spoof".** ⟨|startoftranscript|⟩ ⟨|lt|⟩ ⟨|transcribe|⟩ ⟨|notimestamps|⟩ **1**
VAD	Google SC v2 + Freesound	✗	background	⟨|**nospeech**|⟩	**Identify if there is speech in the provided audio. The options are 0."no speech", 1."speech".** ⟨|startoftranscript|⟩ ⟨|**nospeech**|⟩
AuC	ESC-50	✗	audio_class:17	⟨|**audio**|⟩ ⟨|**auc**|⟩ ⟨|**esc50**|⟩ ⟨|notimestamps|⟩ **audio_class:17**	**Specify the type of environmental noise in the audio. The options are 0."audio_class:17", ⋯** ⟨|startoftranscript|⟩ ⟨|audio|⟩ ⟨|transcribe|⟩ ⟨|notimestamps|⟩ **0**

Table 1: Example prompts for different task types, datasets, and languages, with both single-token task specifiers and natural language phrases as discussed in section 4.1 and 4.2 respectively. The prediction of the model is **bolded**.

tasks, respectively. We experiment with the use of a single token as specifier S^{task} for each task type, S^{lang} for each language, and S^{data} for each dataset as shown in Fig. 1 to represent S^r (Eq. 1):

$$S^r = (S^{\text{task}}, S^{\text{lang}}, S^{\text{data}}) \tag{2}$$

We then propose incorporating human-interpretable natural phrases to enhance the attractiveness of our model for human interaction as shown in the example in Fig. 1. We use a probabilistic formulation to provide a new training objective, which handles various natural phrase prompts. Let I^r be the natural language instruction describing downstream task r. We can model posterior distribution in Eq. 1 as:

$$P(Y^r|X, S^r) = \sum_{I^r} P(Y^r|X, \cancel{S^r}, I^r) P(I^r|\cancel{X,} S^r), \tag{3}$$

where we assume conditional independence of $Y^r|X, I^r$ from S^r and $I^r|S^r$ from X to simplify Eq. 3. With the sampling approximation, Eq. 3 is rewriten as

$$P(Y^r|X, S^r) = \sum_{I^r \sim P(I^r|S^r)} P(Y^r|X, I^r) \tag{4}$$

Instruction I^r serves as samples of diverse natural language task descriptions generated by LLM $P(I^r|S^r)$. By using this training scheme, the model can accept various natural phrase prompts provided by the user as a way of describing S^r (Eq. 2). Unlike prior studies (Chang et al., 2022, 2023), our approach allows for potential generalization to various paraphrases I^r that specify the same task, thereby enhancing the user-friendliness and adaptability of our model for human interaction.

4 Prompt-Based Multi-Tasking

To achieve the formulation described in Eq. 4, this work proposes a prompt-based MTL training format, as illustrated in Fig. 1. The input speech X is passed through encoder-decoder architecture. We prepend instruction I^r to the preceding tokens generated by the decoder (shown as "instruction tuning" in Fig. 1 and described in further detail in section 4.2). The likelihood of the entire label sequence i.e. $P(Y^r|X, I^r)$ in Eq. 4 can be computed using task-specific decoder representations $\mathbf{h}_o^r$ as:

$$P(Y^r|X, I^r) = \prod_{o=1}^{O^r} \text{Softmax}(\text{Out}(\mathbf{h}_o^r)) \tag{5}$$

where Out$(\cdot)$ denotes a linear layer that maps the decoder hidden representation $\mathbf{h}_o^r$ to vocabulary $\mathcal{V}$. The vocabulary $\mathcal{V}$ is the union of label sets $\mathcal{L}^r$ for all the tasks. In this work, both the encoder and decoder are pre-trained using a large speech foundation model (Radford et al., 2022).[2]

4.1 Prompting with task specifier

We provide example prompts using task specifiers (S^{task}, S^{lang}, S^{data} in Eq. 2) for some sample tasks in Table 1. Every decoded text sequence must begin with a ⟨|startoftranscript|⟩ token, which we omit from the example prompts for brevity. The model then performs language identification using S^{lang} similar to its pre-training setup (Radford et al., 2022). In addition to the 99 languages already considered during Whisper pre-training, we add another language tag, ⟨|**audio**|⟩, to enable audio-related tasks such as audio classification. If there is no speech in the audio segment and the model is not prompted using the ⟨|**audio**|⟩ tag, the model has been pre-trained to predict "⟨|nospeech|⟩" and finish decoding, which we finetune for voice activity detection. The next token is the task specifier S^{task}, and we add a new token to the vocabulary of the pre-trained model for each SLU task type. Following the task specifier, we add a new token S^{data} for each dataset since the label set $\mathcal{L}^r$ (see section 3) can differ between datasets. For example, in the case of IC, the label (deactivate_lights_none) in (Lugosch et al., 2019) indicates the same intent as (switchlightoff) in (Saade et al., 2018). Finally, to simplify finetuning, we specify that the model should not predict time stamps. The model then uses these task specifiers to predict the class or label sequence for the given SLU task.

4.2 Prompting with natural language instructions

During training and inference, we query ChatGPT[3] to sample users' expected task descriptions I^r (in Eq. 4) for the given task specifier S^r[4]. We conduct experiments with two settings. In the first setting, we incorporate instructions as previous context text as shown in Fig. 1. For classification tasks, we augment the instructions with a list of possible *options* and make the model predict the option number, such as: "[⟨|startofprev|⟩ **Understanding spoken instructions. The options are 0."go", 1."down", 2."silence",** $\cdots$ ⟨|startoftranscript|⟩ ⟨|en|⟩ ⟨|transcribe|⟩ ⟨|notimestamps|⟩ 1]". In the second setting, we replace the task specifier with the instruction phrase as an ablation study, for example, "[⟨|startoftranscript|⟩ ⟨|en|⟩ **Understanding spoken instructions. The options are 0."go", 1."down", 2."silence",** $\cdots$ ⟨|notimestamps|⟩ 1]". As in section 4.1, we fine-tune the pre-training task types, such as language identification and voice activity detection, similar to the pre-training setup.

We also provide example prompts using natural instructions ("first setting") for some tasks in Table 1. Every decoded text sequence must begin with a ⟨|startofprev|⟩ token, which we omit from the example prompts for brevity. This is followed by natural language instruction followed by a list of possible options. We mask out the training loss over the instruction text, and train the model to predict all other tokens. The next token is ⟨|startoftranscript|⟩ followed by language specifier S^{lang} similar to its pre-training setup (Radford et al., 2022). The next token is the task specifier ⟨|transcribe|⟩ used for ASR during pre-training. Finally, the model predicts the *option number of SLU class* or label sequence for the given SLU task.

5 Experiments

5.1 Datasets and Tasks

We conduct experiments on 18 tasks over 17 publicly available SLU datasets spanning 9 languages and 12 task types, namely IC, named entity recognition (NER), semantic parsing (SP), speech command recognition (SCR), language identification (LID), fake speech detection (FSD), emotion recognition (ER), accent classification (AcC), sarcasm detection (SD), gender identification (GID), voice activity detection (VAD), and audio classification (AuC). We use additional evaluation datasets for testing zero-shot performance in a new dataset and language, as well as for an unseen task type, dialogue act classification (DAC). See section A.2 in the Appendix for task type and dataset description. We will make all our data preparation publicly available for reproduction upon publication.

5.2 Baselines

We compare our approach with SOTA task-specific baselines in prior work (Arora et al., 2022a; Vygon

[2]Although in this work, we only experiment with finetuning Whisper model, our approach can generalize to incorporating any speech foundation model (Peng et al., 2023b).

[3]https://chat.openai.com/

[4]We perform manual inspection to ensure that the generated phrases are human-interpretable task descriptions that a user might provide. More details in section A.5.

Model	FSC (IC)			IEMOCAP (ER)			Google SC v1 (SCR)		
	Seen Prompt	Unseen Prompt	Diff Option Order	Seen Prompt	Unseen Prompt	Diff Option Order	Seen Prompt	Unseen Prompt	Diff Option Order
SOTA *(models trained for each task)*	99.7	✗	✗	79.2	✗	✗	98.6	✗	✗
Task Specifier	99.8	✗	✗	74.7	✗	✗	99.1	✗	✗
Natural Phrase as Prompt	99.7	99.7	99.7	73.9	73.5	74.2	99.1	99.1	99.1
w/ Instruction as Task Specifier	99.7	99.7	99.7	72.8	73.0	73.8	99.1	99.1	99.1
- Options in Instruction	99.7	99.7	✗	61.5	62.0	✗	99.0	99.0	✗

Table 2: Accuracy (%) of MTL model trained using natural language phrases as prompts on FSC, IEMOCAP and Google Speech Commands dataset.

and Mikhaylovskiy, 2021; Tian and Gorinski, 2020; Yen et al., 2021; Bermuth et al., 2022; Shor et al., 2022; Ahamad et al., 2020; Castro et al., 2019; Ray et al., 2022; Hechmi et al., 2021; Jia et al., 2021; Chen et al., 2022) by reporting the performance from the original papers ("SOTA" in Table 3). We also report the performance achieved by SpeechPrompt v2 (Chang et al., 2023) ("SpeechPrompt v2" in Table 3) to quantify the impact of our proposed modifications to prior MTL approaches based on speech prompting, as discussed in Section 3.

5.3 Experimental Setups

All models are built using the ESPNet-SLU toolkit (Arora et al., 2022a). We adopt the evaluation metrics used in prior work.

As a proof of concept, we initially train a MTL model (refer to Section 4.2) employing natural language instructions I^r (in Eq. 4) as prompts for three tasks using the FSC (IC), IEMOCAP (ER), and Google SC v1 (SCR) datasets. We explore various approaches to incorporate natural language phrases, as discussed in Section 4.2. When the list of options is included in the instruction, we introduce a random shuffling of the order of options during training. Our preliminary investigations (not shown) indicate that this shuffling enhances the model's stability across different option orders in the instruction and improves its generalizability to unseen tasks.

Subsequently, we perform multitasking on the same *14* speech classification datasets as SpeechPrompt v2 (Chang et al., 2023), utilizing both a single-token task specifier (Section 4.1) and natural language instructions. We refer to these models as *UniverSLU-14 Task Specifier* and *UniverSLU-14 Natural Phrase*, respectively. We then expand the training tasks to include sequence generation task types such as NER and SP. This enhanced MTL model, trained on a total of *17* datasets, denoted *UniverSLU-17*. To enhance performance on low-resource and para-linguistic datasets, we upsample (see Sec. A.3) the Lithuanian and Arabic speech command datasets, ESC-50, SNIPS, sarcasm detection, and emotion recognition datasets. All model, training, and inference parameters are selected based on validation performance. Further details are provided in section A.3.

6 Results and Discussion

6.1 Using Natural Language Instructions

Table 2 shows the performance of our MTL model trained with natural language phrases instead of task specifiers as discrete prompts. As detailed in Section 4.2, we experiment with various methods of incorporating instructions. We initially report results on seen prompts—natural language instructions encountered during training. In the scenario where the task specifier is replaced with natural instructions, including the list of options in the instruction proves beneficial for performance improvement. Incorporating the natural language instruction as a previous text token yields the best performance, closely resembling that achieved using task specifiers.

To assess the generalizability of our approach, we evaluate the model on natural language phrases not included in training. We average the results obtained from 5 unseen natural language phrases. We ensure that these natural phrases are sufficiently different from training prompts using manual inspection and by making sure that the average edit distance of unseen prompt from training prompts, normalized by the length of the unseen prompt, exceeds 0.9 for all datasets. See the complete list of seen and unseen instructions in Table 9 in the Appendix. Our findings suggest that the model generalizes effectively to unseen natural language phrases, exhibiting performance comparable to that with seen prompts. Additionally, we test the model on two different orders of options in the instruction and report the average performance, noting stable results regardless of option ordering.

This investigation underscores the potential of

Task Type	Metric	Dataset	Language	#Class	SOTA	SpeechPrompt v2	Prompt based MTL			
							UniverSLU-14		UniverSLU-17	
							Task Specifier	Natural Phrase	Task Specifier	Natural Phrase
SCR	Acc ↑	Google SC v1	En	12	98.6 (2021)	94.7	99.1	99.1	99.1	**99.2**
		Grabo SC	Du	36	98.9 (2020)	92.7	**99.7**	**99.7**	**99.7**	**99.7**
		Lithuanian SC	Lt	15	91.8 (2021)	95.5	**100.0**	97.7	98.9	98.9
		Arabic SC	Ar	16	98.9 (2021)	**100.0**	95.9	98.9	**100.0**	**100.0**
IC	Acc ↑	Fluent SC	En	24	99.7 (2022)	98.2	**99.8**	99.7	**99.8**	99.7
	F1 ↑	SNIPS	En	6	96.3 (Tab6)	✗	✗	✗	92.3	63.5
	Acc ↑	SLURP	En	69	89.6 (Tab6)	✗	✗	✗	**90.3**	86.3
LID	Acc ↑	Voxforge	En, Es, Fr, De, Ru, It	6	99.8 (2022)	94.2	**99.9**	**99.9**	**99.9**	**99.9**
FSD	EER ↓	ASVspoof	En	2	2.5 (2022)	13.1	1.0	**0.8**	2.0	0.9
ER	Acc ↑	IEMOCAP	En	4	**79.2** (2022)	50.2	73.4	72.3	74.7	73.9
AcC	Acc ↑	AccentDB	En	9	99.5 (2020)	87.1	**100.0**	99.9	99.9	**100.0**
SD	F1 ↑	MUStARD	En	2	64.6 (2019)	**78.7**	73.2	66.7	73.5	72.9
		MUStARD++	En	2	65.2 (2022)	**75.2**	67.4	66.2	73.6	66.3
GID	F1 ↑	VoxCeleb1	En	2	98.8 (2021)	91.6	**99.9**	**99.9**	**99.9**	**99.9**
VAD	Acc ↑	Google SC v2 + Freesound	En	2	98.8 (2021)	98.3	99.0	99.0	98.8	**99.1**
AuC	Acc ↑	ESC-50	✗	50	**97.0** (2022)	37.5	39.5	2.0	73.0	2.0
NER	SLU F1 ↑	SLURP	En	✗	**79.7** (Tab6)	✗	✗	✗	79.5	74.8
SP	EM ↑	STOP	En	✗	**78.8** (2023)	✗	✗	✗	78.4	75.3

Table 3: Performance of "UniverSLU" employing both task specifiers and natural language instructions on both classification and sequence generation tasks, compared to both task-specific SOTA performance (including Whisper task-specific baselines in Tab. 6 in Appendix) and SpeechPrompt v2 (Chang et al., 2023).

using natural language phrases as prompts. Results support our decision to depart from continuous prompts used in prior work (Chang et al., 2023), opting for natural language instructions to create a more user-friendly model capable of handling diverse paraphrases.

6.2 More tasks: UniverSLU

We present the performance of our "UniverSLU Task Specifier", "UniverSLU Natural Phrase", SpeechPrompt v2, and the SOTA models on these benchmarks in Table 3. Our results demonstrate that the "UniverSLU-14 Task Specifier" outperforms prior speech prompting methods on 11 out of 14 tasks and improves the SOTA on 9 out of 14 tasks. "UniverSLU-14 Natural Phrase" achieves performance comparable to "UniverSLU-14 Task Specifier" across most tasks, with the added benefit of generalizing to new paraphrases of instructions, as discussed in Section 6.1. Our "UniverSLU-17 Task Specifier" model, trained on a combination of speech classification and sequence generation tasks, demonstrates comparable or superior performance compared to the SOTA on 11 tasks. It also achieves competitive performance compared to task-specific baselines on sequence generation tasks, in contrast to previous approaches (Chang et al., 2022) that face challenges in handling sequence generation tasks. We observe that our upsampling approach proves to be highly effective, significantly improving the performance on low-resource datasets such as Arabic speech command recognition. Finally, our "UniverSLU-17 Natural Phrase" model also outperforms SOTA on 10 tasks and again achieves comparable performance to "UniverSLU-17 Task Specifier" model on most tasks.

However, "UniverSLU-14 Natural Phrase" and "UniverSLU-17 Natural Phrase" fail to perform better than the random baseline on the audio classification (AuC) task presumably because Whisper has not been trained on non-speech tasks. Previous work, LTU-AS (Gong et al., 2023a), has showcased the capability of Whisper to follow instructions for performing audio-related tasks. It's important to note that LTU-AS was fine-tuned on a substantial dataset comprising 845,000 audio clips, while our fine-tuning dataset, ESC-50, is notably smaller, consisting of only 1600 training clips. Based on these results, we hypothesize that pre-training plays a crucial role in effective instruction tuning, particularly when dealing with a small fine-tuning dataset.

Comparison to recent LLM-based approaches: As outlined in Section 2, several recent works aim to enhance LLMs with speech processing capabilities. Although the majority of these works do not specifically target SLU tasks, some (Gong et al., 2023a; Tang et al., 2023; Wang et al., 2023) are

Task Type	Metric	Dataset	LauraGPT (2023)	LTU-AS (2023a)	SALMONN (2023)	UniverSLU-17	
			Task Specifier	Natural Phrase	Natural Phrase	Task Specifier	Natural Phrase
IC	Acc ↑	SLURP	87.87	✗	✗	**90.3**	86.3
ER	Acc ↑	IEMOCAP	✗	65.2	69.0	**74.7**	73.9
GID	F1 ↑	VoxCeleb1	✗	90.8	✗	**99.9**	**99.9**
AuC	Acc ↑	ESC-50	✗	**80.8**	✗	73.0	2.0
NER	SLU F1 ↑	SLURP	73.45	✗	✗	**79.5**	74.8

Table 4: Results comparing *UniverSLU* with recently proposed LLMs for speech processing.

Model	Dataset	Language	Task Type
	Snips	KSU Emotions	Daily Talks
	IC	ER	DAC
	F1 ↑	Acc ↑	Acc ↑
SOTA *(supervised models)*	96.3	86.8	61.1
Random	19.7	25.0	25.0
Majority	6.4	28.5	45.5
UniverSLU-14 Natural Phrase	44.6	38.8	17.6
UniverSLU-17 Natural Phrase	✗	41.7	19.3

Table 5: Performance of MTL models trained using natural language phrases as prompts on an unseen dataset, language and task type in a zero-shot manner. We include random and majority baselines for comparison and report the performance of supervised models (Tab6 for SNIPS, 2021 for KSU Emotions, 2023 for DAC).

fine-tuned on a subset of SLU tasks considered in this work, and we show their comparison with UniverSLU in Table 4[5]. UniverSLU-17 outperforms all of these approaches on these tasks, except for AuC where LTU-AS excels due to fine-tuning on a larger audio dataset. We also use SALMONN's publicly available checkpoint (Tang et al., 2023) and report its zero-shot performance on other UniverSLU tasks in section A.4, where it is observed to lag behind UniverSLU models. In conclusion, our MTL approach exhibits substantial advancements over task-specific models as well as previous MTLs, including models that deploy LLM.

6.3 Zero Shot Performance

To assess the generalization capabilities of our multi-task model, we initially evaluate its performance on unseen tasks that assess its ability to handle seen task types in new datasets and languages. For this analysis, we employ the SNIPS and KSU_Emotions[6] (Arabic Emotion Recognition) datasets. During inference, similarly to the training process, we include the possible label options in the instruction prompts. We also show the performance of random and majority baselines, obtained by randomly (uniformly) selecting answers and selecting the most common answer, respectively. Our findings in Table 5 indicate that "UniverSLU-14 Natural Phrase" consistently outperforms random and majority baselines on these tasks. However, a notable performance gap exists when compared to supervised topline models, suggesting substantial room for improvement. Furthermore, we explore the model's capability to perform a completely new task type—Dialogue Act Classification (DAC)—that was not encountered during training. We observe in Table 5 that Univer-SLU models currently fail to generalize to this unseen task type. We also calculate the *following rate* proposed in (Tang et al., 2023), which measures the percentage that the instruction-tuned model can successfully follow the instructions. We observe that it is 100% for "UniverSLU" on all tasks, indicating that it always predicts one of the label options. This preliminary investigation provides insight into the model's generalization ability to unseen tasks in a zero-shot manner.

7 Conclusion

We introduce an "instruction tuning" approach that utilizes human-interpretable natural language instructions, combined with a list of option labels as prompts, to fine-tune speech foundation models on SLU benchmarks. Our experiments demonstrate the model's ability to understand diverse expressions as instructions during inference. Leveraging this approach, we train a single MTL model named "UniverSLU," capable of performing 12 SLU task types, encompassing both classification and sequence generation tasks. UniverSLU not only provides cost efficiency but also achieves competitive performance compared to task-specific baselines, outperforming SOTA models on several SLU benchmarks. Additionally, we find that the model has some generalization ability on new datasets and languages for seen task types. In future work, we will explore the integration of LLM-based decoder and few shot inference.

[5]SALMONN also reports performance on SLURP; however, it focuses on the slot filling task, which solely predicts slot values. In contrast, the NER results reported in this paper aim to predict both the slot label and its corresponding value.

[6]We consider only 4 emotions for evaluation of all our models as discussed in Section A.2.

Limitations

Our instruction tuning approach relies on incorporating the list of option labels in the prompt. A potential limitation arises if the option list becomes too long, as it may surpass the token limit of the decoder. We plan to delve deeper into this aspect in future work. Additionally, Table 5 illustrates that our "Univer-SLU" model currently encounters challenges in generalizing to new, unseen task types. This may be a limitation of Whisper-style models and could perhaps be addressed with the integration of an LLM-based decoder. It may also be that zero-shot generalization is too challenging for certain unseen tasks, requiring at least a few-shot inference.

Broader Impact and Ethics

With our UniverSLU model, we strive to build a single MTL model capable of doing multiple SLU tasks, which saves compute as we do not need separate models for each task. Our approach effectively uses speech foundational models and hence can perform well without requiring large amounts of task-specific labeled data for instruction tuning. Furthermore, we show that our model can perform well in a zero-shot manner for new datasets and languages, which is very useful to ensure speech technologies are available to a wide range of users across various domains. Additionally, we adhere to the ACL Ethics Policy. Our experiments are based on open-source datasets that are widely used in prior work and there is no violation of privacy in our datasets.

References

Bhuvan Agrawal, Markus Müller, Martin Radfar, Samridhi Choudhary, Athanasios Mouchtaris, and Siegfried Kunzmann. 2020. Tie your embeddings down: Cross-modal latent spaces for end-to-end spoken language understanding. *arXiv preprint arXiv:2011.09044*.

Afroz Ahamad, Ankit Anand, and Pranesh Bhargava. 2020. Accentdb: A database of non-native english accents to assist neural speech recognition. In *LREC*, pages 5351–5358. European Language Resources Association.

Antonios Anastasopoulos and David Chiang. 2018. Tied multitask learning for neural speech translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 82–91. Association for Computational Linguistics.

Siddhant Arora, Siddharth Dalmia, Pavel Denisov, Xuankai Chang, Yushi Ueda, Yifan Peng, Yuekai Zhang, Sujay Kumar, Karthik Ganesan, Brian Yan, et al. 2022a. ESPnet-SLU: Advancing spoken language understanding through espnet. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7167–7171. IEEE.

Siddhant Arora, Siddharth Dalmia, Brian Yan, Florian Metze, Alan W Black, and Shinji Watanabe. 2022b. Token-level sequence labeling for spoken language understanding using compositional end-to-end models. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5419–5429, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Siddhant Arora, Hayato Futami, Shih-Lun Wu, Jessica Huynh, Yifan Peng, Yosuke Kashiwagi, Emiru Tsunoo, Brian Yan, and Shinji Watanabe. 2023. A study on the integration of pipeline and E2E SLU systems for spoken semantic parsing toward STOP quality challenge. *CoRR*, abs/2305.01620.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Emanuele Bastianelli, Andrea Vanzo, Pawel Swietojanski, and Verena Rieser. 2020. SLURP: A spoken language understanding resource package. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*. Association for Computational Linguistics.

Lina Benamer and Osama Alkishriwo. 2020. Database for arabic speech commands recognition.

Daniel Bermuth, Alexander Poeppel, and Wolfgang Reif. 2022. Finstreder: Simple and fast spoken language understanding with finite state transducers using modern speech-to-text models. *CoRR*, abs/2206.14589.

Carlos Busso, Murtaza Bulut, Chi-Chun Lee, Abe Kazemzadeh, Emily Mower, Samuel Kim, Jeannette N. Chang, Sungbok Lee, and Shrikanth S. Narayanan. 2008. IEMOCAP: interactive emotional dyadic motion capture database. *Lang. Resour. Evaluation*, 42(4):335–359.

Xingyu Cai, Jiahong Yuan, Renjie Zheng, Liang Huang, and Kenneth Church. 2021. Speech emotion recognition with multi-task learning. In *Interspeech 2021, 22nd Annual Conference of the International Speech Communication Association, Brno, Czechia, 30 August - 3 September 2021*, pages 4508–4512. ISCA.

Rich Caruana. 1997. Multitask learning. *Machine Learning*, 28.

Santiago Castro, Devamanyu Hazarika, Verónica Pérez-Rosas, Roger Zimmermann, Rada Mihalcea, and Soujanya Poria. 2019. Towards multimodal sarcasm detection (an _obviously_ perfect paper). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 4619–4629. Association for Computational Linguistics.

William Chan, Daniel S. Park, Chris A. Lee, Yu Zhang, Quoc V. Le, and Mohammad Norouzi. 2021. Speechstew: Simply mix all available speech recognition data to train one large neural network. *CoRR*, abs/2104.02133.

Kai-Wei Chang, Wei-Cheng Tseng, Shang-Wen Li, and Hung-yi Lee. 2022. SpeechPrompt: An exploration of prompt tuning on generative spoken language model for speech processing tasks. In *Interspeech 2022, 23rd Annual Conference of the International Speech Communication Association, Incheon, Korea, 18-22 September 2022*, pages 5005–5009. ISCA.

Kai-Wei Chang, Yu-Kai Wang, Hua Shen, Iu-thing Kang, Wei-Cheng Tseng, Shang-Wen Li, and Hung-yi Lee. 2023. SpeechPrompt v2: Prompt tuning for speech classification tasks. *CoRR*, abs/2303.00733.

Ke Chen, Xingjian Du, Bilei Zhu, Zejun Ma, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. 2022. HTS-AT: A hierarchical token-semantic audio transformer for sound classification and detection. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2022, Virtual and Singapore, 23-27 May 2022*, pages 646–650. IEEE.

Shijie Chen, Yu Zhang, and Qiang Yang. 2021. Multi-task learning in natural language processing: An overview.

Zhehuai Chen, He Huang, Andrei Andrusenko, Oleksii Hrinchuk, Krishna C Puvvada, Jason Li, Subhankar Ghosh, Jagadeesh Balam, and Boris Ginsburg. 2023. Salm: Speech-augmented language model with in-context learning for speech recognition and translation. *arXiv preprint arXiv:2310.09424*.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*.

Yunfei Chu, Jin Xu, Xiaohuan Zhou, Qian Yang, Shiliang Zhang, Zhijie Yan, Chang Zhou, and Jingren Zhou. 2023. Qwen-audio: Advancing universal audio understanding via unified large-scale audio-language models. *arXiv preprint arXiv:2311.07919*.

José Guilherme Camargo de Souza, Matteo Negri, Elisa Ricci, and Marco Turchi. 2015. Online multitask learning for machine translation quality estimation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 219–228. The Association for Computer Linguistics.

Yassir Fathullah, Chunyang Wu, Egor Lakomkin, Junteng Jia, Yuan Shangguan, Ke Li, Jinxi Guo, Wenhan Xiong, Jay Mahadeokar, Ozlem Kalinli, et al. 2023a. Prompting large language models with speech recognition abilities. *arXiv preprint arXiv:2307.11795*.

Yassir Fathullah, Chunyang Wu, Egor Lakomkin, Junteng Jia, Yuan Shangguan, Jay Mahadeokar, Ozlem Kalinli, Christian Fuegen, and Mike Seltzer. 2023b. Towards general-purpose speech abilities for large language models using unpaired data. *arXiv preprint arXiv:2311.06753*.

Eduardo Fonseca, Jordi Pons, Xavier Favory, Frederic Font, Dmitry Bogdanov, Andres Ferraro, Sergio Oramas, Alastair Porter, and Xavier Serra. 2017. Freesound datasets: A platform for the creation of open audio datasets. In *ISMIR*, pages 486–493.

Heting Gao, Junrui Ni, Kaizhi Qian, Yang Zhang, Shiyu Chang, and Mark Hasegawa-Johnson. 2022. Wavprompt: Towards few-shot spoken language understanding with frozen language models. In *Interspeech 2022, 23rd Annual Conference of the International Speech Communication Association, Incheon, Korea, 18-22 September 2022*, pages 2738–2742. ISCA.

Yuan Gong, Alexander H. Liu, Hongyin Luo, Leonid Karlinsky, and James R. Glass. 2023a. Joint audio and speech understanding. *CoRR*, abs/2309.14405.

Yuan Gong, Hongyin Luo, Alexander H. Liu, Leonid Karlinsky, and James R. Glass. 2023b. Listen, think, and understand. *CoRR*, abs/2305.10790.

Khaled Hechmi, Trung Ngo Trong, Ville Hautamäki, and Tomi Kinnunen. 2021. Voxceleb enrichment for age and gender recognition. In *IEEE Automatic Speech Recognition and Understanding Workshop, ASRU 2021, Cartagena, Colombia, December 13-17, 2021*, pages 687–693. IEEE.

Chien-yu Huang, Ke-Han Lu, Shih-Heng Wang, Chi-Yuan Hsiao, Chun-Yi Kuan, Haibin Wu, Siddhant Arora, Kai-Wei Chang, Jiatong Shi, Yifan Peng, Roshan S. Sharma, Shinji Watanabe, Bhiksha Ramakrishnan, Shady Shehata, and Hung-yi Lee. 2023. Dynamic-superb: Towards A dynamic, collaborative, and comprehensive instruction-tuning benchmark for speech. *CoRR*, abs/2309.09510.

Zhiqi Huang, Milind Rao, Anirudh Raju, Zhe Zhang, Bach Bui, and Chul Lee. 2022. MTL-SLT: Multi-task learning for spoken language tasks. In *Proceedings of the 4th Workshop on NLP for Conversational AI*, pages 120–130, Dublin, Ireland. Association for Computational Linguistics.

Fei Jia, Somshubra Majumdar, and Boris Ginsburg. 2021. Marblenet: Deep 1d time-channel separable convolutional neural network for voice activity detection. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2021, Toronto, ON, Canada, June 6-11, 2021*, pages 6818–6822. IEEE.

Suyoun Kim, Takaaki Hori, and Shinji Watanabe. 2017. Joint CTC-attention based end-to-end speech recognition using multi-task learning. In *Proc. ICASSP*, pages 4835–4839. IEEE.

Aliaksei Kolesau and Dmitrij Šešok. 2020. Unsupervised pre-training for voice activation. *Applied Sciences*, 10(23).

Cheng-I Jeff Lai, Zhiyun Lu, Liangliang Cao, and Ruoming Pang. 2023. Instruction-following speech recognition. *CoRR*, abs/2309.09843.

Kushal Lakhotia, Evgeny Kharitonov, Wei-Ning Hsu, Yossi Adi, Adam Polyak, Benjamin Bolte, Tu Anh Nguyen, Jade Copet, Alexei Baevski, Adelrahman Mohamed, and Emmanuel Dupoux. 2021. Generative spoken language modeling from raw audio. *CoRR*, abs/2102.01192.

Egor Lakomkin, Chunyang Wu, Yassir Fathullah, Ozlem Kalinli, Michael L Seltzer, and Christian Fuegen. 2023. End-to-end speech recognition contextualization with large language models. *arXiv preprint arXiv:2309.10917*.

Keon Lee, Kyumin Park, and Daeyoung Kim. 2023. Dailytalk: Spoken dialogue dataset for conversational text-to-speech. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Bo Li, Ruoming Pang, Tara N Sainath, Anmol Gulati, Yu Zhang, James Qin, Parisa Haghani, W Ronny Huang, Min Ma, and Junwen Bai. 2021. Scaling end-to-end models for large-scale multilingual asr. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1011–1018. IEEE.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.*, 55(9):195:1–195:35.

Loren Lugosch, Mirco Ravanelli, Patrick Ignoto, Vikrant Singh Tomar, and Yoshua Bengio. 2019. Speech model pre-training for end-to-end SLU. In *Interspeech 2019, 20th Annual Conference of the International Speech Communication Association, Graz, Austria, 15-19 September 2019*. ISCA.

Ken MacLean. 2018. Voxforge. `http://www.voxforge.org/home`.

Soumi Maiti, Yifan Peng, Shukjae Choi, Jee-weon Jung, Xuankai Chang, and Shinji Watanabe. 2023. Voxtlm: unified decoder-only models for consolidating speech recognition/synthesis and speech/text continuation tasks. *arXiv preprint arXiv:2309.07937*.

Ali Hamid Meftah, Mustafa A. Qamhan, Yasser Mohammad Seddiq, Yousef Ajami Alotaibi, and Sid-Ahmed Selouani. 2021. King saud university emotions corpus: Construction, analysis, evaluation, and comparison. *IEEE Access*, 9:54201–54219.

Rafael Müller, Simon Kornblith, and Geoffrey E Hinton. 2019. When does label smoothing help? *Advances in neural information processing systems*, 32.

Andreas Nautsch, Xin Wang, Nicholas W. D. Evans, Tomi H. Kinnunen, Ville Vestman, Massimiliano Todisco, Héctor Delgado, Md. Sahidullah, Junichi Yamagishi, and Kong Aik Lee. 2021. Asvspoof 2019: Spoofing countermeasures for the detection of synthesized, converted and replayed speech. *IEEE Trans. Biom. Behav. Identity Sci.*, 3(2):252–265.

Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. 2019. Specaugment: A simple data augmentation method for automatic speech recognition. In *Interspeech*, pages 2613–2617.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Proc. NeurIPS*, 32.

Puyuan Peng, Brian Yan, Shinji Watanabe, and David Harwath. 2023a. Prompting the hidden talent of web-scale speech models for zero-shot task generalization. *arXiv preprint arXiv:2305.11095*.

Yifan Peng, Jinchuan Tian, Brian Yan, Dan Berrebbi, Xuankai Chang, Xinjian Li, Jiatong Shi, Siddhant Arora, William Chen, Roshan Sharma, et al. 2023b. Reproducing whisper-style training using an open-source toolkit and publicly available data. *arXiv preprint arXiv:2309.13876*.

Jonas Pfeiffer, Ivan Vulic, Iryna Gurevych, and Sebastian Ruder. 2020. MAD-X: an adapter-based framework for multi-task cross-lingual transfer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 7654–7673. Association for Computational Linguistics.

Gueorgui Pironkov, Stéphane Dupont, and Thierry Dutoit. 2016. Multi-task learning for speech recognition: an overview. In *ESANN*.

Rohit Prabhavalkar, Takaaki Hori, Tara N. Sainath, Ralf Schlüter, and Shinji Watanabe. 2023. End-to-end speech recognition: A survey. *CoRR*, abs/2303.03329.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. Robust speech recognition via large-scale weak supervision. *CoRR*, abs/2212.04356.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Anupama Ray, Shubham Mishra, Apoorva Nunna, and Pushpak Bhattacharyya. 2022. A multimodal corpus for emotion recognition in sarcasm. In *LREC*, pages 6992–7003. European Language Resources Association.

Vincent Renkens, Steven Janssens, Bart Ons, Jort F. Gemmeke, and Hugo Van hamme. 2014. Acquisition of ordinal words using weakly supervised NMF. In *2014 IEEE Spoken Language Technology Workshop, SLT 2014, South Lake Tahoe, NV, USA, December 7-10, 2014*, pages 30–35. IEEE.

Paul K Rubenstein, Chulayuth Asawaroengchai, Duc Dung Nguyen, Ankur Bapna, Zalán Borsos, Félix de Chaumont Quitry, Peter Chen, Dalia El Badawy, Wei Han, Eugene Kharitonov, et al. 2023. Audiopalm: A large language model that can speak and listen. *arXiv preprint arXiv:2306.12925*.

Alaa Saade, Alice Coucke, Alexandre Caulier, Joseph Dureau, Adrien Ball, Théodore Bluche, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, and Maël Primet. 2018. Spoken language understanding on the edge. volume abs/1810.12735.

Joel Shor, Aren Jansen, Wei Han, Daniel S. Park, and Yu Zhang. 2022. Universal paralinguistic speech representations using self-supervised conformers. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2022, Virtual and Singapore, 23-27 May 2022*, pages 3169–3173. IEEE.

Siddharth Sigtia, Erik Marchi, Sachin Kajarekar, Devang Naik, and John Bridle. 2020. Multi-task learning for speaker verification and voice trigger detection. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2020, Barcelona, Spain, May 4-8, 2020*, pages 6844–6848. IEEE.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

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. *CoRR*, abs/2310.13289.

Yusheng Tian and Philip John Gorinski. 2020. Improving end-to-end speech-to-intent classification with reptile. In *Interspeech 2020, 21st Annual Conference of the International Speech Communication Association, Virtual Event, Shanghai, China, 25-29 October 2020*, pages 891–895. ISCA.

Paden Tomasello, Akshat Shrivastava, Daniel Lazar, Po-Chun Hsu, Duc Le, Adithya Sagar, Ali Elkahky, Jade Copet, Wei-Ning Hsu, Yossef Mordechay, Robin Algayres, Tu Anh Nguyen, Emmanuel Dupoux, Luke Zettlemoyer, and Abdelrahman Mohamed. STOP: A dataset for Spoken Task Oriented Semantic Parsing. In *CoRR*.

Shubham Toshniwal, Hao Tang, Liang Lu, and Karen Livescu. 2017. Multitask learning with low-level auxiliary tasks for encoder-decoder based speech recognition. *arXiv preprint arXiv:1704.01631*.

Roman Vygon and Nikolay Mikhaylovskiy. 2021. Learning efficient representations for keyword spotting with triplet loss. In *SPECOM*, volume 12997 of *Lecture Notes in Computer Science*, pages 773–785. Springer.

Jiaming Wang, Zhihao Du, Qian Chen, Yunfei Chu, Zhifu Gao, Zerui Li, Kai Hu, Xiaohuan Zhou, Jin Xu, Ziyang Ma, Wen Wang, Siqi Zheng, Chang Zhou, Zhijie Yan, and Shiliang Zhang. 2023. Lauragpt: Listen, attend, understand, and regenerate audio with GPT. *CoRR*, abs/2310.04673.

Pete Warden. 2018. Speech commands: A dataset for limited-vocabulary speech recognition. *CoRR*, abs/1804.03209.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022. Finetuned language models are zero-shot learners. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Lianwei Wu, Yuan Rao, Haolin Jin, Ambreen Nazir, and Ling Sun. 2019. Different absorption from the same sharing: Sifted multi-task learning for fake news detection. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 4643–4652. Association for Computational Linguistics.

Yongxin Yang and Timothy M. Hospedales. 2015. A unified perspective on multi-domain and multi-task learning. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Hao Yen, Pin-Jui Ku, Chao-Han Huck Yang, Hu Hu, Sabato Marco Siniscalchi, Pin-Yu Chen, and Yu Tsao. 2021. A study of low-resource speech commands recognition based on adversarial reprogramming. *CoRR*, abs/2110.03894.

Wenyi Yu, Changli Tang, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun Ma, and

Chao Zhang. 2023. Connecting speech encoder and large language model for asr. *arXiv preprint arXiv:2309.13963*.

Yu Zhang, Wei Han, James Qin, Yongqiang Wang, Ankur Bapna, Zhehuai Chen, Nanxin Chen, Bo Li, Vera Axelrod, Gary Wang, Zhong Meng, Ke Hu, Andrew Rosenberg, Rohit Prabhavalkar, Daniel S. Park, Parisa Haghani, Jason Riesa, Ginger Perng, Hagen Soltau, Trevor Strohman, Bhuvana Ramabhadran, Tara N. Sainath, Pedro J. Moreno, Chung-Cheng Chiu, Johan Schalkwyk, Françoise Beaufays, and Yonghui Wu. 2023. Google USM: scaling automatic speech recognition beyond 100 languages. *CoRR*, abs/2303.01037.

Yu Zhang and Qiang Yang. 2017. A survey on multi-task learning. *CoRR*, abs/1707.08114.

Sendong Zhao, Ting Liu, Sicheng Zhao, and Fei Wang. 2019. A neural multi-task learning framework to jointly model medical named entity recognition and normalization. In *AAAI*, pages 817–824. AAAI Press.

Xianrui Zheng, Chao Zhang, and Philip C. Woodland. 2022. Tandem multitask training of speaker diarisation and speech recognition for meeting transcription. In *Interspeech 2022, 23rd Annual Conference of the International Speech Communication Association, Incheon, Korea, 18-22 September 2022*, pages 3844–3848. ISCA.

Jinfeng Zhuang and Yu Liu. 2019. Pintext: A multitask text embedding system in pinterest. In *KDD*, pages 2653–2661. ACM.

A Appendix

A.1 Comparison to Task-Specific Baseline

Since most of these baselines do not incorporate Whisper, we train "task-specific" baselines (see Table 6) by finetuning Whisper on selected datasets, following the approach in (Arora et al., 2023), to better understand the efficacy of training multiple tasks simultaneously. Table 6 presents the results of task-specific baselines and the MTL model ("UniverSLU-3") trained with task specifiers (Section 4.1) on selected speech classification and sequence generation tasks, namely IC, NER, and SP tasks. We start by comparing the task-specific baseline, which involves finetuning the Whisper model, with end-to-end SLU models in ESPnet-SLU (Arora et al., 2022a) (rows 1 and 2). Results show that finetuning Whisper is beneficial for SLU tasks and yields strong baselines.

MTL framework "UniverSLU-3" that incorporates task specifiers $S^{\text{lang}}, S^{\text{task}}, S^{\text{data}}$ (Eq. 2) achieves competitive performance to the task-specific baselines[7]. We additionally investigate training the MTL model using only S^{task} (w/ Task Specifier in Table 3) (e.g., "⟨|**en**|⟩ ⟨|**scr**|⟩ ⟨|notimestamps|⟩ down") and both the S^{task} and S^{data} prompts (w/ Data Specifier in Table 3)(e.g., "⟨|**en**|⟩ ⟨|**scr**|⟩ ⟨|**google_scr**|⟩ ⟨|notimestamps|⟩ down"). We observe similar performance with and without using the dataset specifier (S^{data} in Eq. 2) as an additional prompt. Based on these findings, we conclude that the multi-tasking model achieves comparable performance to task-specific baselines for both classification and sequence generation tasks. We can also significantly reduce the number of trainable parameters since we now have a *single* MTL model instead of multiple models, making it cost-efficient.

A.2 Data and Task Type

Intent Classification (IC) aims to recognize the intent from a user's command in order to take appropriate action. We leverage 2 publicly available corpora, i.e., SNIPS and FSC. The **SNIPS** corpus (Saade et al., 2018) (CC-BY-SA license) consists of 1,660 conversations with in-house voice assistant. For our experiments, we utilize a random split following the approach outlined in (Agrawal et al., 2020). The **Fluent Speech Commands**

[7] Except for the SNIPS dataset where the results appear to be slightly unstable due to the limited size of the test set. It includes only 166 utterances.

Model	SNIPS	SLURP		STOP
	IC F1 ↑	IC Acc ↑	NER SLU F1 ↑	SP EM ↑
Task-specific baselines *(models tuned for each task)*				
ESPnet-SLU	91.7	87.4	77.7	73.3
Whisper task-specific	**96.3**	89.6	79.7	**78.8**
UniverSLU-3 *(single model covers all tasks)*				
w/ Task Specifier S^{task}	93.9	**90.5**	**80.5**	78.4
w/ Dataset Specifier S^{data}	92.7	90.4	80.4	78.3

Table 6: Results showing performance of prompt based MTL model "UniverSLU-3" trained using task specifiers on selected datasets.

(FSC) dataset (Lugosch et al., 2019) (Public License) benchmarks IC performance with 30,043 commands spoken to an intelligent home assistant.

Named Entity Recognition (NER) refers to the task of labeling the spoken tokens in a user's command with associated entities and recognizing mentions of these entities. It is a *sequence generation* task. The **SLURP** corpus (Bastianelli et al., 2020) (CC BY-NC 4.0 license) is an open source NER corpus consisting of 57.4 hours of single-turn conversations with a home assistant, making it relevant to commercial SLU applications. To augment our training set, we incorporate synthetic data, following prior work (Arora et al., 2022a).

Semantic Parsing (SP) is a *sequence generation* task that focuses on converting a spoken utterance into a semantic parse sequence, enabling the voice assistant to execute tasks effectively. The **STOP** dataset (Tomasello et al.) (STOP Dataset License Agreement[8]) is the largest semantic parsing dataset. It comprises over 200,000 audio files from more than 800 speakers across 8 different domains.

Speech Command Recognition (SCR) refers to the task of detecting when a keyword has been spoken in an utterance. The **Google Speech Commands** dataset (Warden, 2018) (CC BY 4.0 license) consists of nearly 100,000 utterances of spoken keywords. The **Grabo** dataset (Renkens et al., 2014) (License for use in academic research or commercial product) comprises English and Dutch spoken commands given to a robot for navigation. The dataset includes recordings from 11 speakers issuing 36 different commands, each repeated 15 times. The **Lithuanian SC** dataset (Kolesau and Šešok, 2020) (license for use in academic research) consists of recordings of 28 individuals uttering 20 Lithuanian words using a mobile phone. The **Arabic SC** database (Benamer and Alkishriwo, 2020) (CC BY 4.0 license) consists of 1,600 speech recordings from 40 speakers in Arabic, covering 6 control words and the digits 0 through 9.

Language Identification (LID) aims to recognize the language of a spoken utterance. **VoxForge** (MacLean, 2018) (GPL license) is an open-source dataset established to collect transcribed multilingual speech using open-source engines. We utilize this dataset to train a language classification system for several European languages, including English, Spanish, French, German, Russian, and Italian. Consistent with previous work (Chang et al., 2023), we use random split with 1200 audio samples in the training, 300 in the validation, and 300 in the test set for each language.

Fake Speech Detection (FSD) aims to detect fake or spoofed speech generated by Text-to-Speech and Voice Conversion systems from real speech in order to prevent adversarial attacks. **ASVSpoof** (Nautsch et al., 2021) (ODC-By license) is a benchmark designed for building fake speech detection systems. In our experiments, we utilize the Logical Access (LA) portion of the ASVspoof 2019 dataset, which includes both bonafide and spoofed speech. The dataset consists of a total of 121,461 utterances with real speech collected from 107 English speakers.

Emotion Recognition (ER) aims to understand the emotion or sentiment conveyed in a utterance. **IEMOCAP** (Busso et al., 2008)[9] is a dataset consisting of approximately 12 hours of speech data with four emotion classes: neutral, happy, sad, and angry. We use Sessions 1-4 as training and Session 5 as a test set.

Accent Classification (AcC) aims to classify speech from speakers with different accents within the same language. **Accent DB** (Ahamad et al., 2020) (CC BY-NC 4.0 License) dataset contains approximately 20 hours of English speech data, encompassing 4 non-native accents, 1 metropolitan Indian accent, and 4 native accents.

Sarcasm Detection (SD) aims to identify whether a given utterance is sarcastic or not, based on verbal and non-verbal cues. The **MUStARD** (Castro et al., 2019) (MIT License) dataset is a multimodal sarcasm detection dataset consisting of 1,991 utterances compiled from popular TV shows and annotated with sarcasm labels. The **MUStARD++** (Ray et al., 2022) (Publicly avail-

[8] https://github.com/facebookresearch/spoken_task_oriented_parsing/blob/main/LICENSE

[9] License:https://sail.usc.edu/iemocap/Data_Release_Form_IEMOCAP.pdf

Tasks Type	Dataset	Language	Output Type	Example Outputs (Y^r)
SCR	Google SC v1	En	speech command class	yes, down, no, $\cdots$
	Grabo SC	Du		turn_rel throttle="slow" angle="east", turn_rel throttle="slow" angle="west", turn_rel throttle="fast" angle="south", $\cdots$
	Lithuanian SC	Lt		ačiū, iki, išjunk, $\cdots$
IC	Arabic SC	Ar	intent class	A, B, C, $\cdots$
	Fluent SC	En		increase heat washroom, deactivate lights, deactivate lights bedroom, $\cdots$
	SNIPS	En		increase brightness, set light color, set light brightness, $\cdots$
	SLURP	En		music likeness, music settings, play music, $\cdots$
LID	Voxforge	En, Es, Fr, De, Ru, It	langauge class	En, Es, Fr, De, Ru, It
FSD	ASVspoof	En	synthetic speech class	spoof, bonafide
ER	IEMOCAP	En	emotion class	angry, neutral, sad, happy, other
AcC	AccentDB	En	speaker accent class	american, australian, bangla,$\cdots$
SD	MUStARD	En	sarcasm class	sarcasm, not sarcasm
	MUStARD++	En		sarcasm, not sarcasm
GID	VoxCeleb1	En	speaker gender class	male, female
VAD	Google SC v2 + Freesound	En	voice activity class	speech, no speech
AuC	ESC-50	✗	audio class	dog, pouring water, breathing, $\cdots$
NER	SLURP	En	sequence of (entity phrase, entity tag) pairs	sl:date FILL tuesday SEP sl:event_name FILL meeting, $\cdots$
SP	STOP	En	sematic parse sequence	[in:create_alarm [sl:date_time_recurring for six am every day]], $\cdots$

Table 7: Overview of the fine-tuning datasets and task types in the UniverSLU model. Each row indicates a "separate" task. Note that a task refers to a specific combination of task type and dataset. For instance, SNIPS and FSC have the same task type but are different tasks due to differences in their label set $\mathcal{L}^r$. We also show the example outputs Y^r (seperated by ",") for each task.

able) dataset is an enhanced version of MUStARD dataset. It includes original MUStARD dataset and features enhanced annotations. Additionally, the dataset has been doubled in size, comprising 2,695 utterances.

Gender Identification (GID) aims to determine the gender of a speaker based on their speech. For gender identification, we utilize the **VoxCeleb1** (Hechmi et al., 2021) (Creative Commons Attribution-ShareAlike 4.0 International License) dataset. It contains approximately 148,000 training and 5,000 test utterances. The gender labels are extracted from the speaker metadata.

Voice Activity Detection (VAD) systems aim to determine whether a given audio contains human speech or background noise. We leverage the **Google Speech Commands v2** (Warden, 2018) (CC BY 4.0 license) dataset as speech data and **Freesound** (Fonseca et al., 2017) (GNU Affero General Public License v3.0) dataset as background noise to train these systems.

Audio Classification (AuC) is a multi-class single-label classification task that aims to correctly classify the audio present in the environment. **ESC-50** (Fonseca et al., 2017) (Creative Commons Attribution-NonCommercial license) is a labeled set of 2000 environment recording with 50 audio classes.

Zero-shot evaluation datasets: To assess the performance of our MTL model in an unseen language, we employ the Arabic Emotion Recognition dataset, KSU_emotions (Meftah et al., 2021) (License: LDC User Agreement for Non-Members). We utilize a subset of this dataset that mirrors the emotions present in the IEMOCAP dataset—namely, neutral, happy, sad, and angry—comprising 2301 utterances. Additionally, we extend our evaluation to a new, unseen task

type: **Dialogue Act Classification (DAC)** which aims to identify the function of a given speech utterance in a dialog. For this task, we utilize the Daily Talks (Lee et al., 2023) (CC-BY-SA 4.0 license) spoken dialogue dataset, annotated with four dialogue act categories—question, inform, directive, and commissive—comprising 4758 utterances, following the setup outlined in (Huang et al., 2023).

Finally, it is important to note that all the datasets are *publicly available*.

A.3 Experimental Setups

Our models are implemented in PyTorch (Paszke et al., 2019), and experiments are conducted with ESPnet-SLU toolkit (Arora et al., 2022a).

Task-specific baselines: We finetune the Whisper medium model with more frequent saving of checkpoints (i.e., after every 1,000 iterations instead of after each epoch) following the approach in (Arora et al., 2023). We train for 100 epochs with a learning rate of 1e-5 and 1500 warmup steps. Early stopping is performed if the validation accuracy saturates. Additionally, we add special tokens to the vocabulary of the Whisper model for SLU labels, such as slot and intent tags. Similar to prior work (Arora et al., 2022a), we use an auxiliary ASR objective to train our SLU models when we have access to transcripts.

MTL models using task specifiers: Similar training setup is followed for training our multitasking model with task specifiers S^{lang}, S^{task}, S^{data} (Eq. 2). In addition to SLU tags, we also add S^{task}, S^{data}, and "⟨|audio|⟩" tokens to the Whisper vocabulary, which are used as prompts for the model, as discussed in Section 4.1. For "UniverSLU-17 Task Specifier" we increase the number of iterations per epoch to 3,000 and train for 100 epochs, since training data becomes larger due to the combination.

MTL models using natural language instruction: For the proof of concept model, trained using 3 datasets, namely FSC, IEMOCAP and Google SC v1, we get 15 natural language descriptions from ChatGPT and use these descriptions to generate instructions for each spoken utterance. We further perform random shuffling of the order of options and combine each natural language description with 3 different random option orders to generate training utterances. For "UniverSLU-14 Natural Phrase" and "UniverSLU-17 Natural Phrase", we query ChatGPT to obtain ten unique natural language task descriptions from ChatGPT. Each description is further combined with 2 random option orders to generate training utterances. We require only a reduced number of task descriptions and option orders since the training data is much bigger than the one for the proof of concept model. Since the number of training utterances is much larger than one for "UniverSLU-14 Task Specifier" (nearly 20 times), we increased the number of iterations per epoch to 25,000 and observed the model to converge in 25 epochs. In contrast to prior approaches, we only add "⟨|audio|⟩" token to the Whisper vocabulary, which is used to fine-tune on audio classification tasks.

Hyperparameter	Value
Whisper model	medium
Dropout Rate	0
LR schedule	inv. sqrt.
Max learning rate	[1e-5, 2e-5, 5e-5]
Warmup steps	[5000, 15000, 25000]
Number of steps	[25, 50, 100]
Adam eps	1e-6
Adam betas	(0.9, 0.99)
Weight decay	0.01
Beam Size	[1, 5, 20]
Length Penalty	[0,0.1,0.2]
Maxlen ratio	[1.0,1.2]

Table 8: Model Training and Inference Search for UniverSLU Models.

The combined parameter size for all our MTL and task-specific models is approximately 762.3M. We initialize the "*UniverSLU-17 Task Specifier*" and "*UniverSLU-17 Natural Phrase*" models with weights obtained from the MTL model "UniverSLU-14 Task Specifier" and "UniverSLU-14 Natural Phrase," respectively. We additionally perform upsampling on low-resource datasets: the Arabic speech commands dataset by a factor of 6, the Lithuanian speech commands dataset, ESC-50, SNIPS, and sarcasm detection datasets by a factor of 3, and the emotion recognition dataset by a factor of 2. We chose these upsampling ratios roughly in inverse proportion to the number of utterances in the datasets. In future work, we will further investigate various data sampling approaches and their impact on model performance.

We apply SpecAugment (Park et al., 2019) and use dropout (Srivastava et al., 2014) and label smoothing (Müller et al., 2019) techniques. The models are trained using 4 NVIDIA A40 (40GB) GPUs. All model, training, and inference parame-

Task Type	Split	Natural Language Phrase
IC	Train/Dev	Assessing and categorizing the meaning behind spoken utterances Uncovering the intent behind verbal communication Sorting and classifying the purpose of spoken statements Decoding and categorizing the intention behind spoken language Understanding and labeling the intent in spoken expressions Organizing and categorizing the meaning of spoken utterances Interpreting and classifying the purpose of verbalized intentions Assigning labels to classify the intent of spoken communication Analyzing and categorizing the underlying purpose of spoken words Grouping and identifying the intention behind spoken utterances Differentiating and classifying the intent expressed through speech Extracting and labeling the classification of spoken intentions Segregating and classifying the purpose of verbalized expressions Unraveling and categorizing the intended meaning of spoken statements Intent classification of spoken utterance
	Test	Classifying the purpose of verbal expression Identifying the intent conveyed through speech Decipher the objective conveyed through speech Determine the purpose of the spoken expression Identify the motive embedded in speech
SCR	Train/Dev	Recognizing speech command Decoding oral requests Grasping speech guidance Discerning spoken triggers Acknowledging vocal commands Comprehending voiced instructions Interpreting verbalized prompts Detecting spoken signals Capturing voice-activated orders Classifying speech-based commands Inferring uttered guidance Analyzing vocal instructions Differentiating voiced prompts Sorting vocalized guidance Parsing spoken directives
	Test	Understanding spoken instructions Identifying uttered directives Identify spoken instructions Understand voiced directives Decode vocal orders
ER	Train/Dev	Emotion recognition of spoken utterance Understanding the emotional content of spoken utterance Interpreting and categorizing emotions in verbal communication Decoding the emotional aspect of spoken expressions Discerning and recognizing emotions in spoken language Assessing and labeling the emotional state of spoken utterances Grasping the emotional intent behind spoken words Uncovering and categorizing the emotional response in speech Extracting and identifying the emotions conveyed through spoken utterances Emotionally analyzing and categorizing spoken expressions Capturing and recognizing emotional cues in spoken utterance Processing and discerning emotions in verbalized communication Differentiating and classifying the emotional tone of spoken utterances Unraveling and categorizing the intended emotional expression of spoken statements Inferring and organizing the classification of emotions in spoken language
	Test	Detecting emotional content in spoken utterance Identifying emotions conveyed through speech Perceive the emotion conveyed through speech Discern the feelings embedded in spoken utterance Categorize the emotional content of verbal expression

Table 9: Detail of Natural Language Phrases used as Prompts to train and evaluate our MTL model in Table 2

ters are selected based on validation performance. Table 8 shows training and inference hyperparameters for our hyperparameter search. Full Details about models, configuration files, and data preparation will be made publicly available prior to publication.

A.4 Detailed Comparison with LLMs

In the evaluation process, we utilized the publicly available SALMONN (Tang et al., 2023) 13B checkpoint and applied it to unseen UniverSLU tasks in a zero shot manner, employing instructions similar to those used in UniverSLU as prompts. The results, however, are presented for a subset of tasks due to various challenges encountered during the evaluation process. Difficulties were faced in obtaining results for sequence prediction tasks (Tomasello et al.), primarily because the output did not match the expected label format. GPU memory constraints were encountered for classification tasks (Lugosch et al., 2019; Renkens et al., 2014) with more than 20 labels since the option labels were included in the instruction. Voice activity detection task with very long audios posed a challenge, leading to out of memory error on a 40 GB GPU.

Table 10 shows that SALMONN 13B did not perform well on most tasks. However, it's crucial to recognize that this comparison with Univer-SLU is not entirely fair, as SALMONN is operating in a zero-shot manner. Notably, SALMONN exhibited even poorer performance than random for accent classification, intent classification (SNIPS) and sarcasm detection tasks. It was also unable to perform arabic speech command recognition, never predicting one of the label options. Furthermore, in zero-shot inference on SNIPS, SALMONN performed worse than UniverSLU-14 Natural Phrase (See Tab. 5). These results strongly support our assertion that existing LLMs struggle with the challenges posed by diverse SLU tasks. The need for UniverSLU, as demonstrated by its superior performance in handling a wide array of SLU tasks, becomes evident in light of these findings.

A.5 Generating Instructions from ChatGPT

In this work, we utilize ChatGPT to generate natural language task descriptions I^r for each task type. We list our prompts for obtaining task descriptions for each task type in Table 11. We further perform manual inspection to ensure that these task descriptions resemble how a user might describe a task. We keep querying the ChatGPT until we have sufficient task descriptions for training and inference of our models.

Task	Metric	Dataset	Language	#Class	SALMONN	UniverSLU-17	
					Natural Phrase	Task Specifier	Natural Phrase
SCR	Acc ↑	Google SC v1	En	12	64.6	99.1	**99.2**
		Lithuanian SC	Lt	15	18.2	**98.9**	**98.9**
		Arabic SC	Ar	16	✗	**100.0**	**100.0**
IC	F1 ↑	SNIPS	En	6	10.9	**92.3**	63.5
LID	Acc ↑	Voxforge	En, Es, Fr, De, Ru, It	6	30.6	**99.9**	**99.9**
FSD	EER ↓	ASVspoof	En	2	19.9	2.0	**0.9**
AcC	Acc ↑	AccentDB	En	9	8.7	99.9	**100.0**
SD	F1 ↑	MUStARD	En	2	37.2	**73.5**	72.9
		MUStARD++	En	2	36.4	**73.6**	66.3
GID	F1 ↑	VoxCeleb1	En	2	98.6	**99.9**	**99.9**

Table 10: Performance of *SALMONN* using publicly available checkpoint on unseen SLU tasks, that were used for fine-tuning *UniverSLU* models.

Task Type	Prompt
SCR	Different ways of saying "recognize speech command" in 15 ways in a natural language without changing meaning
IC	Different ways of saying "classify intent of spoken utterance" in 15 ways in a natural language without changing meaning
LID	Different ways of saying "identifying the language in verbal communication." in 15 ways in a natural language without changing meaning
FSD	Different ways of saying "differentiate speech synthesis and voice conversion speech from real speech" in 15 ways in a natural language without changing meaning
ER	Different ways of saying "emotion recognition of spoken utterance" in 15 ways in a natural language without changing meaning
AcC	Different ways of saying "identifying the accent in spoken utterance" in 15 ways in a natural language without changing meaning
SD	Different ways of saying "detect whether given speech has sarcasm" in 15 ways in a natural language without changing meaning
GID	Different ways of saying "detect gender of speaker in given speech" in 15 ways in a natural language without changing meaning
VAD	Different ways of saying "detect if there is speech in the given audio" in 15 ways in a natural language without changing meaning
AuC	Different ways of saying "classify the environment sound present in audio" in 15 ways in a natural language without changing meaning
NER	Different ways of saying "recognise the named entities in spoken utterance" in 15 ways in a natural language without changing meaning
SP	Different ways of saying "write the semantic parse of spoken utterance" in 15 ways in a natural language without changing meaning
DAC	Different ways of saying "classify the dialog act of spoken utterance" in 15 ways in a natural language without changing meaning

Table 11: Prompts for ChatGPT to generate task descriptions for each task type.