Interspeech 2025
17-21 August 2025, Rotterdam, The Netherlands

QUADS: QUAntized Distillation Framework for Efficient Speech Language Understanding

Subrata Biswas, Mohammad Nur Hossain Khan*, Bashima Islam*

Worcester Polytechnic Institute, USA

{sbiswas, mkhan, bislam}@wpi.com

Abstract

Spoken Language Understanding (SLU) systems must balance performance and efficiency, particularly in resource-constrained environments. Existing methods apply distillation and quantization separately, leading to suboptimal compression as distillation ignores quantization constraints. We propose QUADS, a unified framework that optimizes both through multi-stage training with a pre-trained model, enhancing adaptability to low-bit regimes while maintaining accuracy. QUADS achieves 71.13% accuracy on SLURP and 99.20% on FSC, with only minor degradations of up to 5.56% compared to state-of-the-art models. Additionally, it reduces computational complexity by 60–73× (GMACs) and model size by 83–700×, demonstrating strong robustness under extreme quantization. These results establish QUADS as a highly efficient solution for real-world, resource-constrained SLU applications.

Index Terms: Quantization, knowledge distillation, multi-stage training, speech-language understanding.

1. Introduction

Spoken Language Understanding (SLU), a critical component of Natural Language Understanding, aims to extract semantic information from user utterances [1] and intent detection is one of the key subtasks of SLU, involving classifying the overall purpose of an utterance. As Augmented Reality (AR), Virtual Reality (VR), and voice-assisted technologies continue to proliferate, SLU has become a cornerstone of conversational AI, enabling systems to interpret and derive meaning from user input [2, 3, 4]. With the growing ubiquity of these technologies in everyday life—from virtual assistants like Alexa and Siri to immersive AR applications—the demand for SLU systems that are not only accurate but also responsive, secure, and efficient has never been greater.

Conventional SLU systems typically follow a two-stage process: first, an Automated Speech Recognition (ASR) module converts spoken audio into text; then, an SLU module analyzes the transcribed text to detect the user's intent [5, 6, 7]. Recent works integrate large language models (LLMs) with ASR for intent classification, yielding promising results [2, 8, 9, 10]. However, these architectures are vulnerable to error propagation, where transcription inaccuracies from the ASR module adversely impact intent classification performance [11, 3, 12, 13, 6]. To address this limitation, researchers have explored end-to-end models that directly classify intent from speech, bypassing the intermediate transcription step [12, 3, 4, 14, 15, 16]. These models achieve high accuracy and are particularly suitable for AR, VR, and voice-assisted devices.

However, the substantial size of these end-to-end models—typically ranging from 75.53 to 2422 MB—poses significant challenges for on-device deployment. As a result, these models are often processed in the cloud, introducing practical issues such as increased latency, elevated energy consumption, and privacy risks. For instance, in AR and VR environments, where immediate feedback is crucial for user immersion, latency can disrupt the experience [17, 18]. Similarly, in voice-assisted devices handling sensitive information, cloud processing raises privacy concerns [19]. Frequent data transmission between devices and the cloud also leads to higher energy consumption, impacting battery life in portable devices [20, 21].

These limitations highlight the pressing need for lightweight, efficient SLU models capable of on-device processing. While model compression techniques like knowledge distillation [22], quantization [23, 24], and pruning [25] have been widely adopted to reduce model size, they often fall short in preserving performance due to their sequential application. Traditional methods typically pre-train models using distillation and then apply quantization, leading to compounded errors and suboptimal compression [26, 27]. Disjoint distillation and quantization stages introduce error propagation, where information loss during distillation compounds during quantization, degrading overall performance. Furthermore, this approach struggles with low-bit quantization, as distilled models are not inherently adapted to extreme precision constraints, resulting in significant quantization errors.

To overcome these challenges, this paper introduces QUADS, a unified QUAntized Distillation framework explicitly designed for Spoken Language Understanding (SLU) tasks. By seamlessly integrating distillation and quantization into a cohesive process, QUADS addresses the limitations of traditional methods and enables efficient, high-performance SLU deployment on resource-constrained devices.

Our contributions tackle three critical challenges in developing efficient SLU models for on-device deployment:

1. Unified Distillation and Quantization for Efficient Compression: Balancing model compression with performance retention is a significant challenge, as reducing model size often leads to degraded accuracy. QUADS mitigates this issue by integrating distillation and quantization into a unified process, preventing the error propagation that occurs when these steps are treated separately. This cohesive framework preserves model performance even under extreme compression, enabling robust SLU in compact models.

2. Adaptability to Low-Bit Quantization: Adapting models to low-bit quantization without compromising intent detection accuracy is another key challenge. QUADS employs a multi-

*These authors contributed equally.

This work is supported by the NSF grant CNS-2347692.

10.21437/Interspeech.2025-532

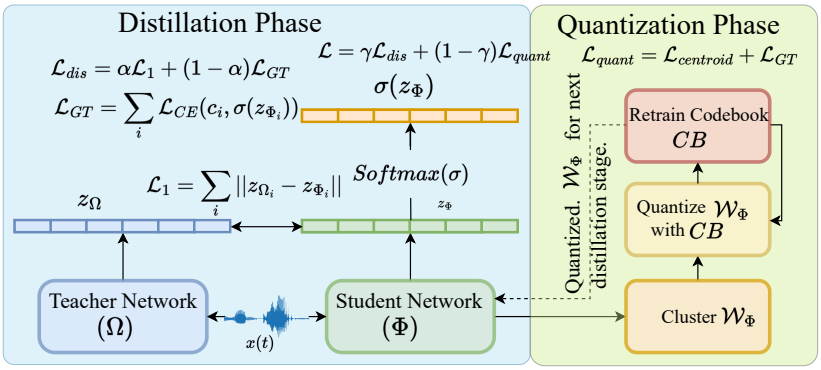


Figure 1: ***Schematic overview of* QUADS.** *A two-phase framework for efficient model training. In the distillation phase, the student model Φ learns from the teacher model Ω via a combined loss strategy. The quantization phase compresses the student model's weights $\mathcal{W}_\Phi$ using the codebook, where weights are grouped into clusters and refined using objectives that balance centroid alignment and cross-network consistency.*

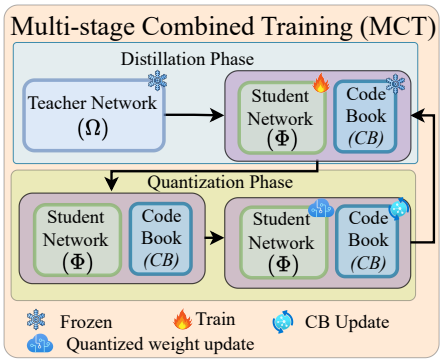


Figure 2: ***Training stages of* QUADS.** *The distillation phase transfers knowledge to the student, and in the quantization phase, the distilled student undergoes quantized weight updates.*

stage combined training strategy that concurrently optimizes the model for both distillation and quantization. This joint optimization enhances the model's adaptability to extreme precision reductions, ensuring high performance even in low-bit regimes.

3. Robust Generalization in Diverse Acoustic Environments: Compressed models often struggle to generalize well across diverse and noisy acoustic environments due to reduced capacity. To address this, QUADS leverages pre-trained acoustic-linguistic representations, enhancing robustness and maintaining high accuracy despite aggressive compression. This ensures consistent performance across varied real-world SLU scenarios.

2. QUADS: Quantized Distillation

To achieve maximum accuracy with minimum computational complexity for a given speech signal $x(t)$ for intent classification, we adopt an Expectation Maximization (EM)[28] approach by integrating model distillation and quantization into a cohesive framework.(Fig.1). Our proposed quantized distillation (QD) framework leverages an iterative multi-stage combined training procedure (MCT) (Fig. 2) to achieve this balance. The following section details critical phases of this process along with the MCT pipeline.

2.1. Distillation Phase

The QD process begins with extracting the mel-spectrogram $X(t,f) = f(x(t))$ from the speech signal $x(t)$, where $f(\cdot)$ represents the mel-spectrogram extraction function. This spectrogram serves as input for both the highly capable, computationally expensive teacher model Ω and the lightweight student network Φ.

The feature representations from the teacher and student networks are denoted as $z_\Omega = f_\Omega(X(t,f)) \in \mathbb{R}^n$ and $z_\Phi = f_\Phi(X(t,f)) \in \mathbb{R}^n$, where $f_\Omega(\cdot)$ and $f_\Phi(\cdot)$ are the respective feature encoders, and n is the latent space size for both networks. To align the student network's feature space z_Φ with the teacher's z_Ω, we compute the l_1 loss:

$$\mathcal{L}_1 = \sum (z_{\Omega_i}, z_{\Phi_i}) ||z_{\Omega_i} - z_{\Phi_i}||_1 \tag{1}$$

A classification head is appended to the student network's encoder $f_\Phi(\cdot)$ to perform intent classification. The cross-entropy loss between the student network's predictions and the ground truth labels is defined as:

$$\mathcal{L}_{GT} = \sum_i \mathcal{L}_{CE}(c_i, \sigma(z_{\Phi_i})) \tag{2}$$

Here, $\sigma(\cdot)$ represents the *softmax* function, and c_i denotes the ground truth labels. The combined distillation loss $\mathcal{L}_{dis}$ balances feature alignment and classification accuracy:

$$\mathcal{L}_{dis} = \alpha\mathcal{L}_1 + (1-\alpha)\mathcal{L}_{GT} \tag{3}$$

2.2. Quantization Phase.

Following distillation, the student model's weights $\mathcal{W}_\Phi$ are quantized using a k-means clustering-based method to further reduce computational complexity. For a given bit length b, $k = 2^b$ centroids are initialized randomly, and the weights $\mathcal{W}_\Phi$ are partitioned into k clusters by minimizing the within-cluster sum of squares: $\arg\min_C \sum_{i=1}^k \sum_{w \in c_i} |w - c_i|^2$

During back-propagation, the gradient of each weight is calculated to update the centroids [29]. The centroid update loss $\mathcal{L}_{centroid}$ is defined using the indicator function $\mathbb{1}(\cdot)$:

$$\frac{\partial \mathcal{L}_{centroid}}{\partial C_k} = \sum_{i,j} \frac{\partial \mathcal{L}_{centroid}}{\partial \mathcal{W}_{\Phi_{i,j}}} \mathbb{1}(I_{i,j} = k) \tag{4}$$

The total quantization loss $\mathcal{L}_{quant}$ is defined as:

$$\mathcal{L}_{quant} = \mathcal{L}_{centroid} + \mathcal{L}_{GT} \tag{5}$$

To effectively integrate both distillation and quantization, we employ a multi-stage training strategy described below.

2.3. Multi-stage Combined Training

To transfer knowledge effectively while maintaining a compact model, we employ a Multi-Stage Combined Training (MCT) strategy, grounded in the principles of Expectation Maximization (EM). MCT alternates between distillation and quantization phases, treating distillation as the expectation step, where the student model learns from the teacher, and quantization as the maximization step, where model parameters are optimized for efficiency. This iterative process progressively refines the student model, as illustrated in Figure 2. The total loss $\mathcal{L}$ is computed by combining the distillation and quantization losses:

Table 1: ***Comparison of* QUADS *and Prior Methods on the SLURP and FSC Datasets.*** *We report accuracy and F1-score for model performance, alongside GMACs and model size, to evaluate efficiency.*

Baseline	Bit Length	SLURP					FSC				
		#Param (M) ↓	Model Size (MB) ↓	GMACs ↓	Accuracy ↑	F1-Score ↑	#Param (M) ↓	Model Size (MB) ↓	GMACs ↓	Accuracy ↑	F1-Score ↑
CTL_{pt}	32	127	484.47	143.2	90.14	82.27	-	-	-	-	-
CTL	32	127	484.47	143.2	72.56	43.34	-	-	-	-	-
Whisper (large)	32	634.94	2422.10	1136.58	75.32	71.11	634.90	2421.97	1136.58	99.49	99.44
Whisper (small)	32	87.05	332.1	172.14	72.16	69.73	87.03	331.97	172.13	99.39	99.31
Whisper (base)	32	19.85	75.73	43.71	71.7	65.48	19.84	75.68	43.70	99.44	99.4
Prosody	32	21.04	80.27	43.82	68.23	62.55	21.04	80.27	44.12	97.80	98.10
Prosody + Distillation	32	21.47	81.92	44.09	76.26	71.92	21.47	81.92	44.31	99.10	98.30
QUADS	32	7.25	27.66	15.6	71.13	65.07	7.64	29.16	18.48	99.20	99.10
	16	7.25	13.83	15.6	70.48	65.21	7.64	14.58	18.48	98.78	98.21
	8	7.25	6.91	15.6	69.73	64.87	7.64	7.29	18.48	98.20	97.87
	4	7.25	3.46	15.6	68.98	64.39	7.64	3.65	18.48	97.39	96.12

$$\mathcal{L} = \gamma \mathcal{L}_{dis} + (1-\gamma)\mathcal{L}_{quant} \tag{6}$$

Here, $\gamma \in \{0, 1\}$ controls the training phase, with $\gamma = 1$ during distillation and $\gamma = 0$ during quantization.

After multiple cycles of distillation and quantization, a final quantization phase is applied. This ensures the model is optimally compressed while maintaining high performance. Unlike intermediate quantization steps, the final phase focuses exclusively on minimizing the model footprint for deployment in resource-constrained environments, solidifying the student model's ability to operate without significant loss in accuracy.

3. Experiments

3.1. Dataset

Following the evaluation of the latest works on intent classification [3, 2, 16], we conduct experiments on two prominent SLU datasets, SLURP and FSC, to ensure comprehensive evaluation across diverse domains and command-specific tasks.

SLURP [30] The dataset comprises 72K 16kHz spoken-language-understanding recordings across 18 distinct domains, split into 49.9K (39.7 h) train, 8.5K (6.8 h) validation, and 12.9K (10.1 h) test utterances.

FSC[31]The dataset consists of 30,000 16kHz single-channel audio recordings of English commands from 97 distinct users designed for smart home and virtual assistant applications.

3.2. State-of-the-Art (SOTA) Baseline Models

We compare QUADS against several SOTA models equipped with either SLU or distillation frameworks [32, 33, 3] while varying model sizes to understand its scalability and efficiency.

Conformer-Transformer-Large (CTL) [32] employs a transformer architecture, leveraging convolutional modules to capture local temporal features and transformer modules to model global dependencies in the audio signal.

Whisper [33] uses convolutional blocks to extract features from the log–mel spectrograms and then passes the feature through a transformer architecture to generate text in an autoregressive manner. We evaluate against the *small*, *base*, and *large* variants.

Prosody [3] leverages prosodic features to generate an attention map for audio over time. We compare against both *prosody-only* and a distillation enhanced *(prosody + distillation)*.

3.3. Evaluation Metrics

To evaluate model performance, we report both *accuracy* and *F1-score* on the test sets. For model efficiency and computational complexity, we report the number of parameters, model size (in megabytes, MB), and the number of multiplication and accumulation operations (GMACs) during inference.

3.4. Implementation Details

We use Whisper *large* as our teacher model Ω. Since the Whisper model accepts mel spectrograms as inputs, we compute an 80-channel log mel spectrogram for all speech samples using 25-millisecond windows with a 10-millisecond stride. Our student model Φ follows a structure similar to that of the teacher. Further implementation details can be found in the open-source codebase [1]. Our student model's encoder is initialized with Whisper pre-trained weights. We add a classifier head after the encoder of Φ for intent classification.

To train the student model, we use learning rates of 1×10^{-6} for the encoder and 1×10^{-3} for the classifier. Our iterative multi-stage training alternates between distillation and quantization for five iterations, with each phase running for epochs.

4. Results

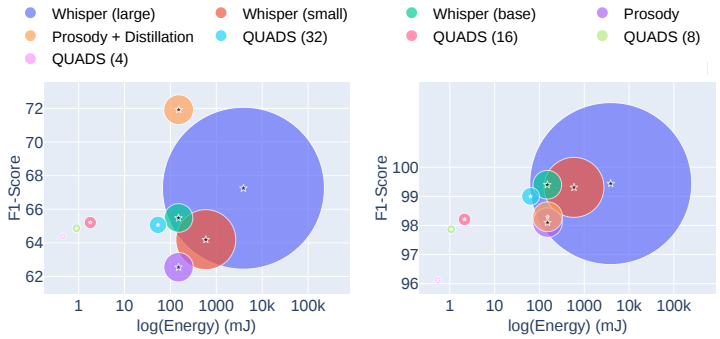
4.1. Comparison with Baseline Algorithms

Table 1 presents a comprehensive comparison between QUADS and SOTA models on the SLURP and FSC datasets. QUADS consistently demonstrates superior efficiency and scalability, making it an ideal candidate for real-world, on-device applications without compromising performance.

SLURP Dataset. On SLURP, QUADS achieves an *F1-score* of 64.39–65.21% and accuracy of 68.98–71.13%, while drastically reducing computational overhead. With a minimal model footprint ranging from 3.46 MB to 27.66 MB and requiring only 15.60 GMACs, QUADS achieves results that are highly competitive with larger, more resource-intensive models.

In contrast, SOTA baselines show marginally higher *F1-scores* of 65.48–71.92% (an average of just 3.9% improvement), but at a significant cost: they demand up to 3× more computational resources (GMACs of 43.72–44.09) and models that are 2.9–23× larger (75.73 MB to 81.92 MB). This highlights QUADS 's unparalleled efficiency, delivering nearly equivalent performance with a fraction of the resource demands. Notably, our 4-bit quantized model occupies only 3.46 MB and contains just 7.25 million parameters while maintaining robust performance, with at most 7.53% drop compared to the Whisper-distilled prosody model. This level of compression, paired with minimal performance degradation, underscores QUADS 's potential for deployment in resource-constrained environments.

[1] https://github.com/BASHLab/QUADS



(a) *SLURP* (b) *FSC*

Figure 3: QUADS *effectively shrinks model size to 3.46–29.16 MB, significantly reducing power consumption—by up to 700× for* ***SLURP*** *and 663× for* ***FSC****—while preserving competitive F1-scores, with a maximum drop of 5.56%. In the visualization, bubble diameter represents model size (MB), and the center of each bubble (⋆) marks the F1-score at a given power consumption.*

Table 2: ***Ablation on model initialization and different training strategy.*** *We study the effect of model initialization and training methods on* QUADS.

Initialization	Bit Length	Training Strategy	SLURP		FSC	
			Model Size (MB) ↓	F1-Score ↑	Model Size (MB) ↓	F1-Score ↑
Random	16	Distillation	78.31	26.59	137.39	88.93
		Quantization after Distillation	48.13	12.91	72.31	85.63
		MCT	13.83	39.78	37.83	90.19
	4	MCT	3.46	29.61	9.4597	89.71
Pre-trained	16	Distillation	78.31	60.93	137.39	98.71
		Quantization after Distillation	48.13	53.79	72.31	96.23
		MCT	13.83	65.21	37.83	98.21
	4	MCT	3.46	64.39	9.4597	96.12

Moreover, while all versions of QUADS maintain consistent GMACs across different bit lengths, the lower bit representations offer significant energy savings [34]. Figure 3a illustrates the energy efficiency versus *F1-score* trade-offs, with bubble sizes representing model sizes in megabytes. Impressively, QUADS consumes 83.29× less energy at the 8-bit level compared to Whisper (base), with only a negligible 3.07% drop in *F1-score*. Such a dramatic reduction in energy consumption solidifies QUADS 's position as a highly efficient alternative to conventional SLU models.

FSC Dataset. On the FSC dataset, QUADS achieves near-perfect *F1-scores* of 99.10–96.12% and accuracies of 99.20–97.39%, closely matching or even surpassing several SOTA baselines while maintaining a significantly smaller model size and lower computational requirements. Compared to the most competitive SOTA models, QUADS requires 83–663× less memory and 61.50× fewer GMACs. Despite these drastic reductions in resource usage, QUADS retains exceptional accuracy, demonstrating that its compact design does not come at the expense of performance.

Figure 3b further illustrates the outstanding trade-off between performance and energy consumption. At the 8-bit representation, QUADS consumes 3637× less energy than Whisper (base) and 141× less energy than Prosody + Distillation, with only a marginal *F1-score* drop of 0.23% and 1.14%, respectively. This remarkable efficiency, coupled with negligible performance degradation, underscores QUADS's suitability for scalable, energy-efficient SLU applications.

4.2. Ablation Study

We conduct an ablation study to examine the influence of model initialization (Random vs. Pre-trained) and training strategies (Distillation, Quantization after Distillation, and MCT) on the performance and efficiency of QUADS. The key findings are presented in Table 2.

Effect of Initialization. Pre-trained initialization consistently outperforms random initialization across all training strategies, underscoring the critical role of leveraging prior knowledge for downstream tasks. On the SLURP dataset, distillation with pre-trained initialization achieves an *F1-score* of 60.93, in stark contrast to 26.59 with random initialization—a remarkable 34.34-point improvement. This trend is even more pronounced on FSC, where all pre-trained models yield *F1-scores* exceeding 96.12, demonstrating superior generalization and robustness. These results highlight that pre-training substantially accelerates convergence and enhances performance, especially for complex, real-world datasets.

Training Strategies. Our results highlight that traditional distillation achieves strong performance (e.g., 98.71 F1 on FSC) but comes with significant computational overhead, resulting in large model sizes (e.g., 137.39 MB). While post-distillation quantization effectively reduces model size (e.g., SLURP: 48.13 MB vs. 78.31 MB), it severely compromises performance, leading to *F1-scores* as low as 12.91. In contrast, our MCT approach harmonizes efficiency and accuracy, delivering the best of both worlds. At 4-bit precision, QUADS compresses models to just 3.46 MB (SLURP) and 9.46 MB (FSC) while maintaining competitive *F1-scores* of 64.39 and 96.12, respectively. This demonstrates that MCT not only mitigates the degradation typically introduced by quantization but also preserves the rich feature representations from the distillation phase, solidifying its superiority in balancing model size and performance.

Bit Length and Dataset Sensitivity. Reducing bit length from 16 to 4 under MCT significantly compresses models without substantial losses in performance. For instance, on SLURP, model size drops from 13.83 MB to 3.46 MB, while *F1-score* remains stable at 64.39%. On FSC, this trend is even more pronounced: pre-trained MCT models at 16-bit precision achieve an outstanding 98.21% F1-score, with minimal decline as bit precision decreases. However, dataset sensitivity varies. The FSC dataset demonstrates remarkable robustness across all configurations, consistently maintaining high *F1-scores* above 96.12. Conversely, SLURP exhibits greater sensitivity to extreme quantization, particularly under 4-bit constraints, suggesting that datasets with more semantic variability may require more careful tuning to maintain peak performance.

5. Conclusion

This study presents a unified distillation and quantization framework that achieves high performance in intent classification with minimal computational overhead. Our model attains *F1-scores* of 64.39–65.07% on SLURP and 96.12–99.10% on FSC, with model sizes as small as 3.46 MB. Compared to state-of-the-art models, QUADS delivers similar accuracy with only a 2–3% drop while significantly reducing memory and energy consumption. These results demonstrate the model's efficiency and suitability for deployment on resource-constrained devices in real-world SLU applications.

6. References

[1] X. Cheng, B. Cao, Q. Ye, Z. Zhu, H. Li, and Y. Zou, "Ml-lmcl: Mutual learning and large-margin contrastive learning for improving asr robustness in spoken language understanding," *arXiv preprint arXiv:2311.11375*, 2023.

[2] Z. Zhu, X. Cheng, H. An, Z. Wang, D. Chen, and Z. Huang, "Zero-shot spoken language understanding via large language models: A preliminary study," in *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, 2024, pp. 17 877–17 883.

[3] S. Rajaa, "Improving end-to-end slu performance with prosodic attention and distillation," in *Interspeech 2023*, 2023, pp. 1114–1118.

[4] X. Zhuang, X. Cheng, and Y. Zou, "Towards explainable joint models via information theory for multiple intent detection and slot filling," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 17, 2024, pp. 19 786–19 794.

[5] E. Kim, A. Jajodia, C. Tseng, D. Neelagiri, T. Ki, and V. R. Apsingekar, "Efficient adaptation of spoken language understanding based on end-to-end automatic speech recognition," in *Interspeech*, 2023.

[6] X. Cheng, Z. Zhu, X. Zhuang, Z. Chen, Z. Huang, and Y. Zou, "Moe-slu: Towards asr-robust spoken language understanding via mixture-of-experts," in *Findings of the Association for Computational Linguistics ACL 2024*, 2024, pp. 14 868–14 879.

[7] X. Cheng, Z. Yao, Z. Zhu, Y. Li, H. Li, and Y. Zou, "C 2 a-slu: cross and contrastive attention for improving asr robustness in spoken language understanding," in *Proc. of INTERSPEECH*, 2023.

[8] J. Hoscilowicz, P. Pawlowski, M. Skorupa, M. Sowański, and A. Janicki, "Large language models for expansion of spoken language understanding systems to new languages," *arXiv preprint arXiv:2404.02588*, 2024.

[9] J. Cho, R. S. Srinivasa, C.-H. Lee, Y. M. Saidutta, C. Yang, Y. Shen, and H. Jin, "Zero-shot intent classification using a semantic similarity aware contrastive loss and large language model," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 10 776–10 780.

[10] L. Fan, J. Pu, R. Zhang, and X.-M. Wu, "Lanid: Llm-assisted new intent discovery," in *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, 2024, pp. 10 110–10 116.

[11] T. Tran, *Neural models for integrating prosody in spoken language understanding*. University of Washington, 2020.

[12] K. Wei, D. Knox, M. Radfar, T. Tran, M. Müller, G. P. Strimel, N. Susanj, A. Mouchtaris, and M. Omologo, "A neural prosody encoder for end-to-end dialogue act classification," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 7047–7051.

[13] S. Wallbridge, P. Bell, and C. Lai, "Do dialogue representations align with perception? an empirical study," in *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, 2023, pp. 2696–2713.

[14] M. Wang, Y. Li, J. Guo, X. Qiao, Z. Li, H. Shang, D. Wei, S. Tao, M. Zhang, and H. Yang, "Whislu: End-to-end spoken language understanding with whisper," in *Proc. Interspeech*, vol. 2023, 2023, pp. 770–774.

[15] E. Kim, Y. Tang, T. Ki, D. Neelagiri, and V. R. Apsingek, "Joint end-to-end spoken language understanding and automatic speech recognition training based on unified speech-to-text pre-training," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 10 971–10 975.

[16] Y. Chen, W. Lu, A. Mottini, L. E. Li, J. Droppo, Z. Du, and B. Zeng, "Top-down attention in end-to-end spoken language understanding," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6199–6203.

[17] C. Kohrs, N. Angenstein, and A. Brechmann, "Delays in human-computer interaction and their effects on brain activity," *PloS one*, vol. 11, no. 1, p. e0146250, 2016.

[18] M. Porcheron, J. E. Fischer, S. Reeves, and S. Sharples, "Voice interfaces in everyday life," in *proceedings of the 2018 CHI conference on human factors in computing systems*, 2018, pp. 1–12.

[19] K. Sun, C. Chen, and X. Zhang, ""alexa, stop spying on me!" speech privacy protection against voice assistants," in *Proceedings of the 18th conference on embedded networked sensor systems*, 2020, pp. 298–311.

[20] A. Alexandridis, K. M. Sathyendra, G. P. Strimel, P. Kveton, J. Webb, and A. Mouchtaris, "Tinys2i: A small-footprint utterance classification model with contextual support for on-device slu," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 7492–7496.

[21] A. Benazir, Z. Xu, and F. X. Lin, "Speech understanding on tiny devices with a learning cache," in *Proceedings of the 22nd Annual International Conference on Mobile Systems, Applications and Services*, 2024, pp. 425–437.

[22] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *International Journal of Computer Vision*, vol. 129, no. 6, pp. 1789–1819, 2021.

[23] J. Wu, C. Leng, Y. Wang, Q. Hu, and J. Cheng, "Quantized convolutional neural networks for mobile devices," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4820–4828.

[24] H. Shao, B. Liu, W. Wang, X. Gong, and Y. Qian, "Dq-whisper: Joint distillation and quantization for efficient multilingual speech recognition," in *2024 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2024, pp. 240–246.

[25] Y. Wang, X. Zhang, L. Xie, J. Zhou, H. Su, B. Zhang, and X. Hu, "Pruning from scratch," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 07, 2020, pp. 12 273–12 280.

[26] Z. Li, Z. Wang, M. Tan, R. Nallapati, P. Bhatia, A. Arnold, B. Xiang, and D. Roth, "Dq-bart: Efficient sequence-to-sequence model via joint distillation and quantization," *arXiv preprint arXiv:2203.11239*, 2022.

[27] W. Zhang, L. Hou, Y. Yin, L. Shang, X. Chen, X. Jiang, and Q. Liu, "Ternarybert: Distillation-aware ultra-low bit bert," *arXiv preprint arXiv:2009.12812*, 2020.

[28] T. Moon, "The expectation-maximization algorithm," *IEEE Signal Processing Magazine*, vol. 13, no. 6, pp. 47–60, 1996.

[29] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding," *arXiv preprint arXiv:1510.00149*, 2015.

[30] E. Bastianelli, A. Vanzo, P. Swietojanski, and V. Rieser, "Slurp: A spoken language understanding resource package," *arXiv preprint arXiv:2011.13205*, 2020.

[31] L. Lugosch, M. Ravanelli, P. Ignoto, V. S. Tomar, and Y. Bengio, "Speech model pre-training for end-to-end spoken language understanding," *arXiv preprint arXiv:1904.03670*, 2019.

[32] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu *et al.*, "Conformer: Convolution-augmented transformer for speech recognition," *arXiv preprint arXiv:2005.08100*, 2020.

[33] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," in *International conference on machine learning*. PMLR, 2023, pp. 28 492–28 518.

[34] E. Ohiri, "Nvidia a100 versus h100: how do they compare?" https://www.cudocompute.com/blog/comparative-analysis-of-nvidia-a100-vs-h100-gpus, 2023, [Accessed 02-08-2025].