IN PURSUIT OF BABEL - MULTILINGUAL END-TO-END SPOKEN LANGUAGE UNDERSTANDING

Markus Müller, Samridhi Choudhary*, Clement Chung, Athanasios Mouchtaris, Siegfried Kunzmann*

Amazon Alexa AI

{mumarkus,samridhc,chungcle,mouchta,kunzman}@amazon.com

ABSTRACT

End-to-end spoken language understanding (E2E SLU) systems predict the utterance semantics directly from speech. So far, to the best of our knowledge, E2E models have only been trained to recognize the semantics for a single language. In this work we introduce the first multilingual E2E SLU system and present results across three languages – English, Spanish and French. We propose a transformer-based, multilingual acoustic encoder to predict intents, that leverages pre-training for both acoustic and linguistic modalities of the SLU model. It learns a robust, cross-modal latent space using a pre-trained multilingual BERT as a semantic teacher. The best performing model achieves relative improvements of 7.2% in a single language setting, 5-6% in two, and 4-6% in three language settings. An intent-wise analysis shows that semantic supervision becomes more important for shorter utterances, while providing an explicit language identifier at the input leads to lower intent classification errors.

Index Terms— spoken language understanding, multilingual, speech recognition, human-computer interaction

1. INTRODUCTION

Spoken language understanding (SLU) is the task of extracting semantics from a spoken utterance [1, 2]. Conventional SLU systems consist of a cascade of two independent modules - an automatic speech recognition (ASR) module to convert the user's speech to text followed by a natural language understanding (NLU) module to convert the transcription into application semantics [3, 4]. This modular approach has been very effective as it allows independent module training using system-specific data that are often available in large quantity. However, training these modules with separate optimization objectives can result in unintended, full-system errors [5]. For example, an ASR system that is trained for the lowest word error rate (WER) would have the same WER for omitting any single word in the sentence - 'turn on the light' . However, omitting 'the' will potentially have little to no impact on the NLU performance, whereas omitting 'on' or 'light' would be an irrecoverable error, for the same WER.

*Equal contribution

Training an end-to-end (E2E) SLU system that predicts the semantics directly from speech is becoming an increasingly popular approach to address the aforementioned limitation [5–15]. E2E SLU models are trained to directly maximize the SLU prediction accuracy. However, these architectures (like any other E2E architecture), require sufficient quantity of high quality speech data with associated semantic labels. In contrast to cascaded systems that leverage both ASR and text-only NLU data for training the individual components, E2E SLU training data is comparatively scarce. This situation is exacerbated for low-resource languages. A variety of methods like knowledge transfer, data augmentation and pre-training have been explored to address this issue [6, 9, 16–19]. One of the most popular approaches is to pre-train model components using available data. In particular, pre-training the network on an ASR task has shown considerable improvements [9, 18, 20]. This idea is further extended in [21] that leverages pre-training for the text-modality along with ASR pre-training in an E2E SLU model. This is done by learning a cross-modal latent space (CMLS) between the text and an acoustic encoder and leads to considerable improvements.

So far, E2E SLU systems have mostly been trained to support one language. Data from different languages have been utilized in an E2E SLU setting for pre-training only, where the goal is to improve the performance of a monolingual model on a different, often low-resource, target language [18, 22]. However, in an increasingly globalized world, it should be possible for users to address their voice assistants (VAs) in the language(s) of their choice. A possible solution is to combine multiple monolingual SLU systems with a component for language identification. While this would solve the problem for some applications, it increases the system resource footprint and requires training and maintaining multiple models. Creating multilingual SLU systems is the key to achieving efficient and more natural interactions with the VAs. These systems not only understand multiple languages but can also be better at capturing non-native pronunciations of multilingual speakers. Moreover, the language agnostic features learned by these models can be leveraged to improve the performance of SLU systems for low-resource languages, thereby helping the monolingual setting as well.

In this work, we propose a transformer-based, multilin-

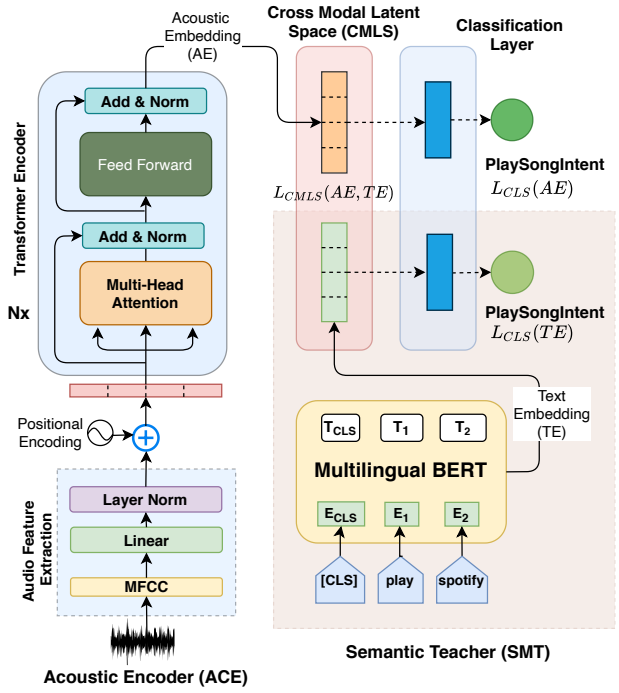

Fig. 1. Cross-Lingual, Cross-Modal Model Architecture

gual E2E SLU system, which to the best of our knowledge is the first multilingual E2E SLU model. Our E2E model predicts the intents directly from the input audio. Supporting multiple languages adds another dimension of complexity to an already sophisticated E2E model that has to learn an optimal state across two modalities - speech and text. We present a principled approach to leverage maximum amount of pre-training across modalities, spanning three languages - English, Spanish and French. Specifically, we extend the work in [21, 23] to learn a cross-lingual CMLS and use a multilingual BERT encoder [24] to provide linguistic semantic supervision. Our approach allows us to not only use, often abundant, text-only data with BERT, but also leverage acoustic features from multiple languages by pre-training the acoustic encoder. Furthermore, we perform an intent-wise analysis of our multilingual models and study the impact of pre-training across different intent classes. Finally, we test the multilingual system performance in a language-informed scenario, where we explicitly pass the language identity (LID) at the input and assess the resulting classification error rates.

2. METHODOLOGY

An E2E SLU model can be seen as a tightly integrated system across two modalities (speech and text) [21, 25], which becomes increasingly data hungry as we add support for multiple languages. To design an efficient multilingual system, we want to leverage maximum possible pre-training across languages to prime the model layers with proper inductive biases. We extend the multi-modal setup of the E2E SLU explored in [21, 23] to a multilingual scenario. This setup not only allows us to leverage ASR and text-only pre-training across languages, but also results in maintainable model footprint, with the bulky text-branch only used during training and removed for inference.

2.1. Model Architecture

Figure 1 shows our proposed architecture. It consists of three components: an *acoustic encoder (ACE)*, a *semantic teacher (SMT)* and a shared *classification layer*. Our ACE is a 768 dimensional, 3-layer transformer encoder with 8 attention heads per layer [26]. The input audio is pre-processed to produce Mel frequency cepstral coefficients (MFCC) features that are passed through a dense layer and normalized before adding a positional encoding to prepare the acoustic input. The encoder output is maxpooled across the time dimension to obtain a fixed-dimensional vector, which we call the *acoustic embedding (AE)*. The AE summarizes the input audio, independent of the length of the input signal.

Similar to [21], we use a pre-trained BERT encoder as the SMT. Since we are dealing with multiple languages, we use a pre-trained multilingual BERT [24], which is first fine-tuned on NLU specific, text-only data, across different languages. The learned BERT encoder is then used as the multilingual SMT, with an explicit embedding loss to learn a common embedding space (L_{CMLS}). The input to the SMT consists of the text transcription of the input audio. We experiment with two ways of extracting the text embedding (TE) from BERT – (1) using the last layer transformer-encoder representation of the CLS token; or (2) pooling the hidden states from the last 4 layers of the transformer encoder [24]. The shared classification layer is a fully-connected network, followed by a softmax to predict the semantic label (intent in our case). It produces an intent prediction using both the AE and TE separately, resulting in computing an embedding specific classification loss $L^{\text{AE}}_{\text{CLS}}$ and $L^{\text{TE}}_{\text{CLS}}$, respectively. The explicit loss to tie the AE and TE is specified by L_{CMLS}. The entire network is trained on a joint loss as shown below –

$$L_{\text{joint}} = L^{\text{AE}}_{\text{cls}} + \lambda_1 L^{\text{TE}}_{\text{cls}} + \lambda_2 L_{\text{CMLS}}, \quad (1)$$

where λ_1 and λ_2 are hyper-parameters that are fine-tuned on a validation set and control the effect of embedding and text classification loss on the network optimization. The embedding loss pushes the acoustic and BERT embeddings closer together. Gradients from the embedding loss (L_{CMLS}) are only propagated to the acoustic branch of the model, with the text branch getting the updates from just the text classification loss. This is primarily done to push the acoustic encoder to encode linguistic information from the BERT based SMT. This teacher has been trained on large amount of text-only data, with appropriate pre-training objectives including

masked language model and semantic tagging. SMT is only used during training to build in semantic robustness into the acoustic encoder. During inference, the teacher portion of the network (everything in the brown box in Figure 1) is removed and the learned ACE is used to make intent predictions using the classification layer. Since the model is trained to learn CMLS with an explicit embedding loss, the latent representations extracted by ACE during inference are semantically richer and should lead to better prediction accuracy.

2.2. Embedding Losses

We experiment with two L_{CMLS} formulations – the 'L_2 loss' as used in [21] and a 'triplet CMLS loss' [23]. Our triplet loss formulation is shown below –

$$L_{\text{CMLS}}(x_a, x_+, x_-) = \max\{0, m + d(x_a, x_+) - d(x_a, x_-)\}$$

x_a is the anchor, x_+ and x_- are the positive and negative examples respectively. We use the AE of the current instance as x_a, TE of an utterance belonging to the same intent class as x_+ and TE of an utterance from a different intent class as x_-. Both x_+ and x_- are selected randomly.

2.3. Acoustic Encoder Pre-training

We pre-train the encoder using the connectionist temporal classification loss (CTC) [27] and use characters as acoustic modeling units. Pre-training ACE to transcribe audio initializes the network to learn helpful features for speech recognition that have proven to increase the classification performance of SLU tasks [9].

2.4. BERT Fine-tuning

We use the pre-trained multilingual BERT encoder from [28] and fine-tune it on NLU tasks. This helps the encoder learn useful representations for our target SLU task and is done by attaching task-specific heads to the BERT encoder to predict the NLU labels – domain, intent and slots, similar to the setup in [29]. For predicting the domain and intent labels, we use one dense layer each to take in the pooled representation of the CLS token, followed by softmax. In order to predict the slots, the last layer hidden state for each input token is passed on to another dense layer to produce a token-level slot label. Post training, the task specific heads are removed and the learned encoder layers are used as our SMT. We produce two BERT encoders – *BERT-IC* and *BERT-Full*. *BERT-IC* contains just the intent classification (IC) head and has a similar target task setup as our proposed model. *BERT-Full* contains domain, intent and slot classification heads in a multi-task setup, where the network is trained on a joint loss composed ofthe individual cross-entropy classification losses.

3. EXPERIMENTAL SETUP

We use a PyTorch [30] based setup[1] for our experiments. We (1) first pre-train ACE using CTC, then (2) fine-tune ACE on IC by replacing the CTC layer with an IC layer. We perform step (2) in two scenarios: with and without semantic supervision using SMT. Our hypothesis is that pre-training on CTC followed by fine-tuning with semantic supervision should lead to the most robust model. When running our experiments, we decided to keep the size of the model constant through different experimental conditions. We aim at building a multilingual system with the same resource footprint as its monolingual counterpart. The experiments are split into two main categories: we first evaluate our pre-training and fine-tuning methods based on a single target language while mixing in data from multiple source languages (used for pre-training only). In the second set of experiments, we add additional languages as target languages to build multilingual systems.

3.1. Model Evaluation

We use the *Intent Classification Error Rate (ICER)* for evaluating our models. This is computed as the ratio of number of incorrect intent predictions to the total number of utterances. For our experiments, we report the relative ICER Reduction (*ICERR*) when compared to the respective baselines. Due to the nature of our dataset, we cannot report absolute baseline ICERs, but all our baselines have an ICER of $< 15\%$ on their respective evaluation sets.

We perform a grid search over learning rates ϵ [1e-5, 1e-4], dropout ϵ [0.1, 0.5], λ_1, λ_2 (equation 1) ϵ [0.2, 1.0] and experiment with different hierarchical unfreezing strategies for SMT as described in [31]. As explained in section 2, we experiment with both *BERT-IC* and *BERT-Full*, with two ways of getting TE[2] across all the aforementioned hyperparameter combinations to get the best performing model across different language combinations.

3.2. SLU Datasets

The corpus used for training our model consists of proprietary real-world, far-field, de-identified VA utterances amounting to about 100h per language. The dataset contains 15 intents with equal number of utterances per intent. We use data from three languages – English (EN), French (FR) and Spanish (ES) – each covering the same set of intents. We further split the dataset into a smaller subset of 50h by randomly selecting utterances to simulate a low-resource scenario. For evaluation, we use a separate dataset with 120h per language and

[1]The source code of the base system is available at `https://github.com/alexa/alexa-end-to-end-slu`, where we will also add code for our proposed model pre-training.

[2][CLS] token representation vs pooled hidden states from last 4 layers.

equal representation of each intent. To extract acoustic features, we use a standard ASR pre-processing pipeline to extract 40 dimensional MFCC features. Extracted with 25ms windows, spaced at 10ms intervals, we applied frame stacking with a context of +/- 1 frames, resulting in 120 dimensional features. We then downsampled these stacked features to a frame rate of 30ms, using only every third frame to feed into the network.

3.3. NLU Datasets

The goal of this dataset is to perform BERT fine-tuning and is different from our SLU Dataset above. We use a random snapshot of proprietary, de-identified VA utterances along with internally crowd-sourced, text-only, NLU data for three languages – EN, ES and FR. We filter the data to contain the same 15 intents per language over 4 domains and > 80 slots. The total number of utterances varies, with each language containing at least 800h of training data. Each dataset instance consists of the utterance text, domain, intent and slot tags.

4. RESULTS

We evaluate our proposed method in a series of experiments. First, we analyze the effectiveness of ACE and SMT pre-training using a single target language. We then expand the number of target languages and train multilingual systems, comparing them to their respective monolingual baselines. We observe that ACE and SMT pre-training result in varying amount of improvements across different intent classes. Therefore, we perform an intent-wise analysis of our multilingual models to study this further. Finally, we test our multilingual system performance in a language-informed scenario, where we explicitly pass the language ID at the input and asses the model performance across languages. For all the experiments, ACE pre-training is performed on the SLU datasets while SMT fine-tuning is done on the NLU datasets (section 3). The final fine-tuning of the E2E model with semantic supervision is done on the SLU datasets. Multilingual datasets are created by combining equal amounts of language specific data. All the results showcased are from our best performing models obtained by performing an extensive grid search as explained in section 3.

4.1. ACE and SMT Pre-training

We study the effectiveness of using acoustic pre-training and semantic supervision, especially by pooling data across languages and compare the model performance across the two formulations of $L_{\text{CMLS}} - L_2$ and triplet loss. In order to limit the number of variables and to have a maintainable number of experiments, we keep the target language (the one that is used to fine-tune the final model) fixed to a single language (ES) for this setup. Languages are incrementally added to the respective ACE and SMT pre-training conditions. Specifically, we first pre-train ACE and SMT with ES, followed by ESEN and finally ESENFR, resulting in three different pre-trained model parts. Once pre-trained, each of these model components are combined in our E2E setup as shown in figure 1 and the final model is fine-tuned on ES SLU data only. The baseline model is a transformer-based ACE, with the same architecture and dimensions as the ACE of our proposed model. It is trained from scratch without any pre-training or semantic supervision on the ES (target language) SLU data.

Table 1. Relative reduction in ICER (ICERR) for pre-training methods (ACE and SMT) using ES as the target language. Pre-training is performed by incrementally pooling data across ES, EN and FR.

Exp.	ACE	SMT	Loss	ES ICERR
Baseline	–	–	–	–
ES1	ES	–	–	4.9
ES2	ESEN	–	–	5.7
ES3	**ESENFR**	**–**	**–**	**6.0**
ES4	ES	ES	L2	4.5
ES5	**ES**	**ES**	**Triplet**	**5.4**
ES6	ESEN	ESEN	L2	4.1
ES7	**ESEN**	**ESEN**	**Triplet**	**6.9**
ES8	ESENFR	ESENFR	L2	4.9
ES9	**ESENFR**	**ESENFR**	**Triplet**	**7.2**

As shown in Table 1, just pre-training ACE improves the ICER by 4.9% (ES1) when compared to the baseline model. Adding a second language increases the improvement (ES2) to 5.7%, indicating that the increased amount of data during pre-training (twice as much for two languages) results in a better initialization of the model. Adding a third language to the pre-training (ES3) improves the performance over pre-training on two languages, indicating that given three times the amount of data across three different languages, the network is able to learn more robust, language independent features.

When SMT supervision is added to the model, the performance depends heavily on the type of L_{CMLS} used. While using the L_2 loss results in $\approx$ 4 - 5% improvement over the baseline model, it is not able to perform better than just having ACE pre-training with no semantic supervision, across all conditions (ES1 vs ES4, ES2 vs ES6 and ES3 vs ES8). However, triplet loss outperforms the L_2 loss across all conditions. More noticeably, combining ACE pre-training and SMT supervision with triplet loss consistently helps across all settings (ES1 vs. ES5, ES2 vs. ES7, and ES3 vs. ES9). The best performing model is the one that has both ACE pre-training and SMT supervision with triplet loss (ES9) with a 7.2% ICERR over the baseline.

Table 2. Relative reduction in ICER (ICERR) of pre-training methods (ACE and SMT) on multilingual models covering two languages (ESEN) and three languages (ESENFR)

Exp.	Fine tuning	ACE	SMT	ES ICERR	EN ICERR	FR ICERR
Baseline	–	–	–	–	–	–
ML1	ESEN	–	–	1.7	1.8	–
ML2	ESEN	ESEN	–	5.2	4.5	–
ML3	ESEN	ESENFR	–	5.6	4.6	–
ML4	ESEN	ESEN	ESEN	5.4	4.8	–
ML5	**ESEN**	**ESENFR**	**ESENFR**	**5.9**	**5.0**	–
ML6	ESENFR	–	–	2.0	2.2	1.6
ML7	ESENFR	ESENFR	–	4.8	4.0	5.1
ML8	**ESENFR**	**ESENFR**	**ESENFR**	**5.5**	**4.7**	**5.9**

4.2. Multilingual SLU

Next we build and evaluate multilingual models. Similar to the previous experiments, we incrementally add languages to our model's training data. However, this time we do that for ACE/SMT pre-training as well as for the final model fine-tuning, resulting in multilingual models supporting two and three languages. For each resulting multilingual model, we compute the ICER for all component-language test sets. The language-specific performance is compared to their respective monolingual baseline models that consists of a transformer-based ACE, similar to the baseline model in Section 4.1, that is trained from scratch on a single (target) language SLU dataset. For the sake of clarity in results and observations, we only report the SMT supervision numbers with triplet loss as it consistently outperforms L_2, as shown in table 1.

In the first set of experiments (ML1 through ML5), we fine-tune the E2E model on the joint ENES data sets, with ACE and SMT pre-training done on both ENES and ENESFR. As shown in Table 2, training on 2 languages in itself (ML1) improves the performance over the monolingual baselines. Pre-training ACE with CTC further improves the performance (ML2). Increasing the number of languages in pre-training (ML2 vs. ML3) improves the performance furthermore. As the system sees more variety during pre-training, the network is initialized in a more robust fashion. Combining ACE pre-training with SMT supervision consistently improves performance, with slightly larger improvement when pre-training on more languages (ML2 vs ML4; ML3 vs ML5).

Next, we add FR to the mix of languages for the final E2E fine-tuning and train the model jointly on three languages (ML6-ML8). Reductions in ICER are observed across all 3 languages with larger improvements over the baseline, even without pre-training (ML1 vs. ML6). Comparing the three language versions with their two language counterparts (ML3 vs. ML7 and ML5 vs. ML8), we see minor degradations in ICER for ES and EN. The primary reason behind this could be that when training a system on three instead of two languages, the network has to approximate three different distributions in the data instead of two. This renders the problem more challenging because the system has to support one additional language. Here too, the best performing model is the one that has both ACE pre-training along with semantic supervision with $\approx$ 5 - 6% improvement over the baseline across all languages.

4.3. Intent-wise Analysis

Table 3. Relative changes in ICER (ICERR) for the ES evaluation set for a multilingual model trained on data from ES, EN FR. While selection of intents is shown, the system is trained on the full data set covering all 15 intents.

Experiment		Time	Music	Set
Average utterance length		4.7s	5.4s	8.8s
Baseline		-	-	-
ESENFR w/ ACE	(ML7)	2.0	0.8	12.3
ESENFR w/ ACE+SMT	**(ML8)**	**6.0**	**1.9**	**17.0**

While there are ICER improvements across all intents in our dataset, the effect of ACE and SMT pre-training varies across each intent class. In order to study this effect further, we note the changes in ICER for three representative intent classes in Table 3 with varying average utterance lengths. We show results based on the systems ML7 and ML8 from Table 2. The 'time' intent contains utterances with queries about the current time, 'music' has queries regarding playing some music and 'set' features queries to set a reminder. 'Time' on average has the shortest utterances with 4.7s. With less acoustic evidence available due to short utterances, semantic pre-training becomes especially important, leading to three times more improvement over just using the ACE pre-training (2% to 6%). For 'music', we see a similar picture with three times more improvement over using just ACE. The 'set' intent has the longest utterances on average (8.8s) with almost double to

Table 4. Relative reduction in ICER if the language information is added to the acoustic features via one-hot encoding. The multilingual system is jointly trained on EN, ES, FR.

Exp.	Fine tuning	ACE + SMT	LID	ES ICERR	EN ICERR	FR ICERR
Baseline	–	–	–	–	–	–
LID1	ESEN	ESEN	–	5.4	4.8	–
LID2	**ESEN**	**ESEN**	✓	**6.1**	**5.5**	–
LID3	ESENFR	ESENFR	–	5.5	4.7	5.9
LID4	**ESENFR**	**ESENFR**	✓	**6.5**	**5.3**	**7.1**

that of the 'time' with 4.7s. Due to the length, more acoustic evidence is available, and hence, the improvement due to SMT supervision is not as pronounced as the other two intents. This indicates that SMT becomes more relevant if less audio is available.

4.4. Language-informed SLU

So far, we trained our models without explicitly providing a language identity to them. While this closely approximates a real-world scenario in which people can talk to a system in any of the languages it was trained on, another mode of operation is to provide the language information explicitly to the network. For this, we encoded the language identity (LID) as one-hot vector and stacked it on top of the acoustic features. In case of e.g. three languages, this LID vector would have a dimensionality of three and would be appended to the acoustic features (say 120 dimensions) resulting in an input vector with a dimensionality of 123. This way, the network has knowledge about the language information associated with each utterance. As shown in Table 4, supplying the LID to the network improves the ICER across all languages and pre-training conditions for both two- and three-language multilingual models. This confirms our hypothesis that adding an explicit language signal to the input audio, helps boost the multilingual model performance.

5. CONCLUSION

We introduced the first multilingual E2E SLU system using data from three languages – English, Spanish and French. Our model architecture is based on cross-modal latent space, with an acoustic and a text-only network component. This network topology allows us to use text-only NLU data for training via a pre-trained multilingual BERT model, which acts as a semantic teacher. The model with the highest improvements uses acoustic pre-training and semantic supervision. Leveraging the full pre-training potential, we reduce the intent classification error rate across different language combinations. Our model achieves improvements of 7.2% with a single target language, 5-6% with two target languages, and 4-6% in a three target language setting. An intent-wise analysis on selected intents shows that the observed gains correlate with the utterance lengths. Providing the language identity explicitly to the encoder further reduces the error rate, which is expected and we provide this numbers for reference. In future, we plan to extend this work to include slot tagging task along with intent classification and scale the system to more languages. Furthermore, we plan on experimenting with different types of semantic teachers that lead to more robust CMLS for an E2E setup.

6. ACKNOWLEDGEMENT

We would like to thank Grant Strimel, Ross McGowan and Nathan Susanj for their insightful discussions and valuable feedback.

7. REFERENCES

[1] Ye-Yi Wang, Li Deng, and Alex Acero, "Spoken language understanding," *IEEE Signal Processing Magazine*, vol. 22, no. 5, pp. 16–31, 2005.

[2] Gokhan Tur and Renato De Mori, *Spoken language understanding: Systems for extracting semantic information from speech*, John Wiley & Sons, 2011.

[3] Cheongjae Lee, Sangkeun Jung, Kyungduk Kim, Donghyeon Lee, and Gary Geunbae Lee, "Recent approaches to dialog management for spoken dialog systems," *Journal of Computing Science and Engineering*, vol. 4, no. 1, pp. 1–22, 2010.

[4] Lin-shan Lee, James Glass, Hung-yi Lee, and Chun-an Chan, "Spoken content retrieval—beyond cascading speech recognition with text retrieval," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 9, pp. 1389–1420, 2015.

[5] Parisa Haghani, Arun Narayanan, Michiel Bacchiani, Galen Chuang, Neeraj Gaur, Pedro Moreno, Rohit Prabhavalkar, Zhongdi Qu, and Austin Waters, "From audio to semantics: Approaches to end-to-end spoken language understanding," in *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 720–726.

[6] Michael Saxon, Samridhi Choudhary, Joseph P. McKenna, and Athanasios Mouchtaris, "End-to-end spoken language understanding for generalized voice assistants," *CoRR*, vol. abs/2106.09009, 2021.

[7] Martin Radfar, Athanasios Mouchtaris, and Siegfried Kunzmann, "End-to-end neural transformer based spoken language understanding," in *Interspeech 2020,th Annual Conference of the International Speech Communication Association*, 2020, pp. 500–505.

[8] Dmitriy Serdyuk, Yongqiang Wang, Christian Fuegen, Anuj Kumar, Baiyang Liu, and Yoshua Bengio, "Towards end-to-end spoken language understanding," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5754–5758.

[9] Loren Lugosch, Mirco Ravanelli, Patrick Ignoto, Vikrant Singh Tomar, and Yoshua Bengio, "Speech model pre-training for end-to-end spoken language understanding," in *Proc. Interspeech 2019*, 2019, pp. 814–818.

[10] Yuan-Ping Chen, Ryan Price, and Srinivas Bangalore, "Spoken language understanding without speech recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 6189–6193.

[11] Yao Qian, Rutuja Ubale, Vikram Ramanaryanan, Patrick Lange, David Suendermann-Oeft, Keelan Evanini, and Eugene Tsuprun, "Exploring asr-free end-to-end modeling to improve spoken language understanding in a cloud-based dialog system," in *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2017, pp. 569–576.

[12] Sahar Ghannay, Antoine Caubrière, Yannick Estève, Nathalie Camelin, Edwin Simonnet, Antoine Laurent, and Emmanuel Morin, "End-to-end named entity and semantic concept extraction from speech," in *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 692–699.

[13] Vincent Renkens and Hugo van Hamme, "Capsule networks for low resource spoken language understanding," in *Proc. Interspeech 2018*, 2018, pp. 601–605.

[14] Marco Dinarelli, Nikita Kapoor, Bassam Jabaian, and Laurent Besacier, "A data efficient end-to-end spoken language understanding architecture," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 8519–8523.

[15] Hong-Kwang J. Kuo, Zoltán Tüske, Samuel Thomas, Yinghui Huang, Kartik Audhkhasi, Brian Kingsbury, Gakuto Kurata, Zvi Kons, Ron Hoory, and Luis Lastras, "End-to-End Spoken Language Understanding Without Full Transcripts," in *Proc. Interspeech 2020*, 2020, pp. 906–910.

[16] Antoine Caubrière, Natalia Tomashenko, Antoine Laurent, Emmanuel Morin, Nathalie Camelin, and Yannick Estève, "Curriculum-based transfer learning for an effective end-to-end spoken language understanding and domain portability," in *Proc. Interspeech 2019*, 2019, pp. 1198–1202.

[17] Natalia Tomashenko, Antoine Caubrière, and Yannick Estève, "Investigating adaptation and transfer learning for end-to-end spoken language understanding from speech," in *Proc. Interspeech 2019*, 2019, pp. 824–828.

[18] Swapnil Bhosale, Imran Sheikh, Sri Harsha Dumpala, and Sunil Kumar Kopparapu, "End-to-end spoken language understanding: Bootstrapping in low resource scenarios," in *Proc. Interspeech 2019*, 2019, pp. 1188–1192.

[19] Pengwei Wang, Liangchen Wei, Yong Cao, Jinghui Xie, and Zaiqing Nie, "Large-scale unsupervised pre-training for end-to-end spoken language understanding," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7999–8003.

[20] Ryan Price, Mahnoosh Mehrabani, and Srinivas Bangalore, "Improved end-to-end spoken utterance classification with a self-attention acoustic classifier," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 8504–8508.

[21] Yinghui Huang, Hong-Kwang Kuo, Samuel Thomas, Zvi Kons, Kartik Audhkhasi, Brian Kingsbury, Ron Hoory, and Michael Picheny, "Leveraging unpaired text data for training end-to-end speech-to-intent systems," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7984–7988.

[22] Ryan Price, "End-to-end spoken language understanding without matched language speech model pretraining data," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7979–7983.

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

[24] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 4171–4186.

[25] Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Ng, "Multimodal deep learning," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, Lise Getoor and Tobias Scheffer, Eds., New York, NY, USA, June 2011, ICML '11, pp. 689–696, ACM.

[26] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, pp. 6000–6010.

[27] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 369–376.

[28] Thomas Wolf, Julien Chaumond, Lysandre Debut, Victor Sanh, Clement Delangue, Anthony Moi, Pierric Cistac, Morgan Funtowicz, Joe Davison, Sam Shleifer, et al., "Transformers: State-of-the-art natural language processing," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2020, pp. 38–45.

[29] Hamidreza Saghir, Samridhi Choudhary, Sepehr Eghbali, Clement Chung, and Amazon Alexa AI, "Factorization-aware training of transformers for natural language understanding on the edge," in *Proc. Interspeech 2021*, 2021.

[30] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala, "PyTorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., pp. 8024–8035. Curran Associates, Inc., 2019.

[31] Jeremy Howard and Sebastian Ruder, "Universal language model fine-tuning for text classification," *arXiv preprint arXiv:1801.06146*, 2018.