ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 979-8-3503-4485-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICASSP48485.2024.10445934

PROBABILITY-AWARE WORD-CONFUSION-NETWORK-TO-TEXT ALIGNMENT APPROACH FOR INTENT CLASSIFICATION

Esaú Villatoro-Tello[⋆,1], *Srikanth Madikeri*[†,1], *Bidisha Sharma*[4], *Driss Khalil*[1], *Shashi Kumar*[1], *Iuliia Nigmatulina*[1,2], *Petr Motlicek*[†,1,3], *Aravind Ganapathiraju*[4]

[1] Idiap Research Institute, Martigny, Switzerland
[2] Institute of Computational Linguistics, University of Zürich, Switzerland
[3]Brno University of Technology, Brno, Czech Republic
[4] Uniphore Software Systems Inc., Palo Alto, CA, USA

ABSTRACT

Spoken Language Understanding (SLU) technologies have greatly improved due to the effective pretraining of speech representations. A common requirement of industry-based solutions is the portability to deploy SLU models in voice-assistant devices. Thus, distilling knowledge from large text-based language models has become an attractive solution for achieving good performance and guaranteeing portability. In this paper, we introduce a novel architecture that uses a cross-modal attention mechanism to extract bin-level contextual embeddings from a word-confusion network (WNC) encoding such that these can be directly compared and aligned with traditional text-based contextual embeddings. This alignment is achieved using a recently proposed tokenwise constrastive loss function. We validate our architecture's effectiveness by fine-tuning our WCN-based pretrained model to do intent classification (IC) on the well-known SLURP dataset. Obtained accuracy on the IC task (81%), depicts a 9.4% relative improvement compared to a recent/equivalent E2E method.

Index Terms— Word-Confusion-Networks, Cross-modal Alignment, Knowledge Distillation, Intent Classification

1. INTRODUCTION

Voice-operated interactive devices rely on Spoken Language Understanding (SLU) to derive the semantics such as slots, intents, and cause-effect signals [1, 2]. With the rising attention from application standpoint, there have been several efforts in the literature, starting from conventional pipeline method that uses automatic speech recognition (ASR) decoded text to derive intent [3], to end-to-end approaches capable of directly extracting semantics from raw speech signal [4–6].

In a conventional pipeline SLU system (i.e., ASR + Natural Language Understanding (NLU)), the ASR system is optimized to minimize the Word Error Rate (WER), while the NLU module is typically trained on clean text. However, the errors in ASR output in fact negatively affect the performance of NLU systems. To reduce the ambiguity caused by ASR errors, previous works utilized ASR 1-best result [7, 8], N-best lists [9, 10], word lattices [11, 12], and word confusion networks (WCNs) [7, 13–15] as inputs to train an SLU system. WCN-based approaches [7, 13, 16] primarily focus on better encoding or explore different modeling approaches to exploit rich information embedded in the WCN.

End-to-end (E2E) SLU methods have substantial potential to alleviate the effects of ASR errors in the pipeline setup [4], however, they require large amounts of training data. Thus, to avoid this restriction recent approaches propose the exploitation of pretrained speech models [17, 18], knowledge distillation by aligning speech and text embeddings by means of a cross-attention layer [6, 19, 20] and the use of multi-modal techniques [5]. Particularly, results reported by [6] are encouraging, however, their proposed approach has not been evaluated on a more challenging dataset (e.g., SLURP). In addition and as our main motivation, there is a gap in the SLU literature regarding the impact of distilling knowledge from text-based embeddings to effectively perform a WCN-to-Text alignment for Intent Classification (IC) task. Hence, this paper is a step towards exploring the impact of a WNC-based representation by implementing a finer embedding alignment technique between WCNs and text-based embeddings.

Overall, our work has three main contributions: (1) a novel architecture to extract bin-level contextual embeddings from WCNs; (2) an exhaustive evaluation of the impact of the WCN-to-Text alignment process for IC task on the SLURP dataset; and (3) the introduction of stronger text-based baselines through optimizing BERT hyperparameters.[1]

[⋆]Corresponding author: *esau.villatoro@idiap.ch*

This work was supported by the Idiap & Uniphore collaboration project.

[†]Partially supported by the EU Horizon 2020 RIA programme under grant agreement No 101007666 / ESPERANTO / H2020-MSCA-RISE-2020, as well as by the Ministry of Education, Youth and Sports of the Czech Republic through the e-INFRA CZ (ID:90254).

[1]Our code is publicly available: `https://github.com/idiap/Word-Confusion-Network-to-Text-Alignment`

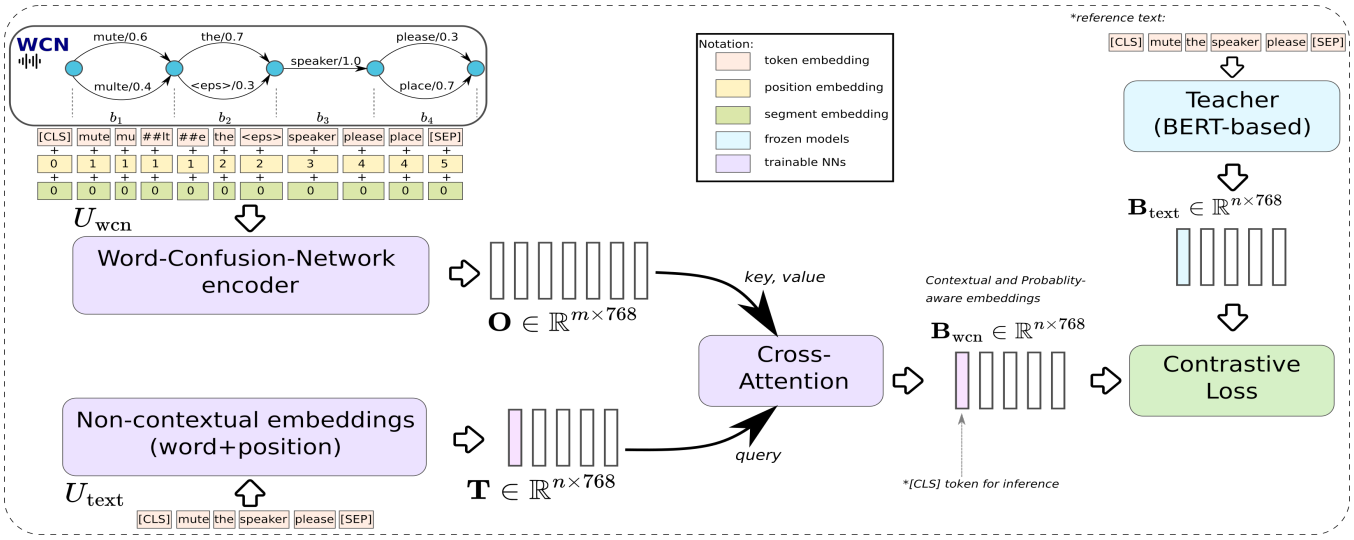

Fig. 1. An overview of the proposed WCN-to-Text alignment architecture. During pretraining, the cross-attention mechanism injects WCNs' contextual, and posterior probabilities information for all the input tokens using the corresponding utterance. During fine-tuning for IC, when only speech information is available in the form of a WCN, only the [CLS] token is required.

2. PROPOSED METHODOLOGY

Figure 1 shows an overview of the proposed WCN-to-Text alignment framework (WCN2B). Let U_{wcn} denote the word confusion network of a spoken utterance and U_{text} its corresponding reference transcription. The WCN encoder takes the lattice structure U_{wcn} as input and returns a representation denoted by a matrix $\mathbf{O} \in \mathbb{R}^{m \times 768}$ where m is the total number of WordPiece tokens present in all the bins from the WCN structure (§2.1). Similar to [6], a Non-Contextual (NC) representation of U_{text} is obtained from a randomly initialized word embedding, which takes a sequence of WordPiece tokens of U_{text} prepended and appended by the `[CLS]` and `[SEP]` tokens, respectively. As these embeddings are not contextual, we also add absolute positional encodings to the output of the NC word embedding. Thus, the NC representation of U_{text} is denoted by a matrix $\mathbf{T} \in \mathbb{R}^{n \times 768}$, where n is equal to the number of WordPiece tokens in the reference text.[2] Then, a cross-modal attention mechanism injects the WCNs' contextual, and probabilities information into $\mathbf{T}$, resulting in $\mathbf{B}^{\text{wcn}}$ embeddings (§ 2.2), which are in turn aligned with BERT via contrastive loss (§ 2.3).

2.1. Word-Confusion-Network Encoder

The WCN is a compact lattice structure where candidate words paired with their associated posterior probabilities are aligned at each position [16, 21]. Formally, given a sequence of word bins $B = (b_1, \ldots, b_M)$, the m-th bin is defined as $b_m = \{(w_m^1, P(w_m^1)), \ldots, (w_m^{I_m}, P(w_m^{I_m}))\}$, where I_m denotes the number of candidates in b_m, w_m^i and $P(w_m^i)$ are the i-th candidate word and its posterior probability respectively.

[2]Notice that $m >> n$ given the nature of the WCN.

Then, for *Word-Confusion-Network encoder* to be able to consume the WCN graph, the WCN is flattened into a word sequence such that: $w^{\text{WCN}} = (w_1^1, \ldots, w_1^{I_1}, \ldots, w_M^1, \ldots, w_M^{I_M})$. Thus, we denote the input sequence as $w = (w_1, \ldots, w_M) =$ `[CLS]` $\oplus$ TOK(w^{WCN}) $\oplus$ `[SEP]`, where $\oplus$ concatenates sequences together, TOK($\cdot$) is the WordPiece tokenizer which tokenizes words into sub-words, `[CLS]` and `[SEP]` are auxiliary tokens for separation, $M = |\text{TOK}(w^{\text{WCN}})| + 2$. Considering the structural characteristics of the WCN, i.e., multiple words competing in a bin, all words and sub-words in the same bin will share the same position ID.

Following the ideas proposed in [16], the WCN encoder consists of a series of bidirectional transformer encoder layers [22], each of which contains a multi-head self-attention module and a feed-forward network with residual connections.[3] Thereby, the input layer of the WCN encoder embeds w into d-dimensional continuous representations, in our case $d = 768$, resulting in a summarized representation $x_t \in \mathbb{R}^d$.

Then, we apply the same extension to the self-attention mechanism to consider the posterior probabilities of tokens in the WCN. Particularly, for each input token sequence $w = (w_1, \ldots, w_M)$, its corresponding probability $p = (p_1, \ldots, p_M)$ is defined as:

$$p_t = \begin{cases} P(w_t) & \text{if } w_t \in \text{TOK}(w^{\text{WCN}}) \\ 1.0 & \text{otherwise,} \end{cases} \quad (1)$$

where $P(w_t)$ represents the ASR posterior probability of token w. Notice that the probability of a sub-word will be equal to that of the original word in the WCN. Now, in order to consider the ASR posterior probabilities in the transformer en-

[3]Reported experiments employed 12 transformer layers, each with 12 attention heads.

coder, the attention value for the l-th layer and the h-th head $e_{ij}^{l,h}$ is computed as follows:

$$e_{ij}^{l,h} = \frac{(W_Q^{l,h} x_i^l)^\intercal (W_K^{l,h} x_j^l)}{\sqrt{d/H}} + \lambda^{l,h} \cdot p_j, \tag{2}$$

where $\lambda^{l,h}$ is a trainable parameter. Finally, token-level representations for each bin contained in the WCN are produced after the stacked encoder layers, resulting in a representation denoted by a matrix $\mathbf{O} \in \mathbb{R}^{m \times 768}$

2.2. Cross-Modal attention

We use the WCN representation $\mathbf{O}$ to inject the ASR's contextual, and posterior probabilities information into the NC embeddings $\mathbf{T}$ such that the resulting contextual embedding implicitly benefits from the WCNs' richness of information. For this, we employ a cross-modal attention mechanism, following a *query-key-value-based* mechanism [22], where the NC embeddings $\mathbf{T}$ act as the query and the WCN embeddings $\mathbf{O}$ act as the keys and values.

Hence, $\mathbf{Q}$, $\mathbf{K}$ and $\mathbf{V} \in \mathbb{R}^{m \times 768}$ are obtained by $\mathbf{Q} = \mathbf{T}\mathbf{W}_q$, $\mathbf{K} = \mathbf{O}\mathbf{W}_k$ and $\mathbf{V} = \mathbf{O}\mathbf{W}_v$, where $\mathbf{W}_q$, $\mathbf{W}_k$ and $\mathbf{W}_v \in \mathbb{R}^{768 \times 768}$ are learnable weights. Now, the contextual and probability-aware embeddings $\mathbf{B}_{\text{wcn}} \in \mathbb{R}^{n \times 768}$ are computed by $\mathbf{B}_{\text{wcn}} = \text{softmax}(\mathbf{Q}\mathbf{K}^\intercal)\mathbf{V}$.

2.3. Contrastive Loss

Once we have the contextual and probability-aware embeddings $\mathbf{B}^{\text{wcn}}$, we align them with the semantically rich BERT contextual representation $\mathbf{B}^{\text{text}}$ on a token-by-token basis as they have the same sequence length n. To achieve this, we apply the recently proposed contrastive loss function ($\mathcal{L}$) described in [6], which facilitates the alignment between sequences pairs of token representations.

Thus, the output sequences in a batch are row-wise concatenated such that $\mathbf{B}^{\text{text}}$ and $\mathbf{B}^{\text{wcn}} \in \mathbb{R}^{s \times 768}$, where s is the sum of all sequence lengths in a batch. Now, the contrastive loss is defined as:

$$\mathcal{L} = -\frac{\tau}{2s} \sum_{i=1}^{s} \left(\log \frac{\exp(c_{ii})}{\sum_{j=1}^{s} \exp(c_{ij})} + \log \frac{\exp(c_{ii})}{\sum_{j=1}^{s} \exp(c_{ji})} \right),$$

where τ is a temperature hyperparameter, and c depicts the cosine similarity between rows i and j in $\mathbf{B}^{\text{text}}$ and $\mathbf{B}^{\text{wcn}}$. As described in [6], the contrastive loss function allows bringing the representation of the same tokens (positive pairs) from the two considered modalities close together and pushes apart different tokens (negative pairs).

For all our reported experiments, we use WCN-text pairs from 1000 hours of People's Speech data [23] for pretraining. Our models were trained on 8 GeForce RTX 3090 GPU following a distributed learning approach during 600k steps using a batch size of 64 utterances, AdamW optimizer with a learning rate of 1e-4, and $\tau = 0.07$.

2.4. Fine-tuning

One main advantage of the proposed architecture is that it does not require having access to utterances' transcriptions, and it can be applicable in a real-life scenario as long as the intermediate WCN representation is available. Thus, we can fine-tune the pre-trained model for the IC downstream task. Particularly, only the learned NC ($\mathbf{T}$) embeddings for the `[CLS]` are used to attend over the WCN encoder output through the cross-attention layer. This allows us to work using a contextual, and probability-aware BERT-like `[CLS]` token for representing the input utterance. For all the performed experiments, the `[CLS]` token is fed to a single linear layer for classification.

For all the IC experiments, the entire model is fine-tuned using the training subset (58 hours) of SLURP dataset [24], with a learning rate of 2e-5 using the AdamW optimizer and a batch size of 32 utterances. WCNs were obtained following the approach described in Section 3.1.

3. EXPERIMENTS AND RESULTS

3.1. ASR+WCN Generation

Following [15], we use XLSR LF-MMI setup as the acoustic model for ASR: the XLSR-53 model [25] is fine-tuned with 390 hours of English data from AMI [26] and Switchboard [27] datasets using the E2E-LFMMI loss function [28, 29] with biphone units [30–32] trained from a graphemic lexicon of size 1M. The Language Model (LM) was trained with 34M utterances from publicly available English datasets including People's Speech, Fisher, Switchboard, AMI, Wikitext103, and subsets of Common Crawl and Reddit datasets. The model was then further fine-tuned with 560 hours of YouTube data with the incremental semi-supervised learning approach with four iterations [33]. Kaldi toolkit [34] is used for Weighted Finite State Transducer (WFST)-based decoding with the default parameter values for beam and lattice beam – 16 and 8, respectively. To adapt the XLSR-53 acoustic model to SLURP, we fine-tuned the model with the train subset of the SLURP data without changing the LM. ASR performances before and after fine-tuning to SLURP are given in Table 1.

To generate WCNs [21], the acoustic scores in the obtained lattices after decoding are scaled followed by minimum Bayes Risk decoding (mBR) [35]. The scaling alleviated the peakiness of the scores. Lattice pruning prior to mBR – the default behavior of Kaldi – was avoided to preserve the richness of the hypotheses in the WCN.

3.2. Baselines

Two baselines were considered: *text*-based (i.e., pipeline) and *acoustic*-based (i.e., E2E). As text-based baseline models, we evaluate the performance of fine-tuning the *bert-base-uncased* pre-trained transformer-based, to which a final linear

Table 1. Alignment accuracy during the pretraining, and ASR performance (in WER) of the XLSR-53 model.

Model	XLSR-53 adaptation	Alignment ACC (↑) People's Speech (*dev*)	SLURP WER (↓) *dev*	SLURP WER (↓) *test*
$S2B_{(XLSR53)}$	✗	74%	34.0	34.4
$WCN2B_{(XLSR53)}$	✗	**77%**		
$S2B_{(XLSR53)}$	✓	72%	16.1	15.5
$WCN2B_{(XLSR53)}$	✓	**75%**		

layer was added to classify the input using the `[CLS]` classification token. To make the baseline as standard and simple as possible we made use of the *Transformers* Python package *AutoModelForSequenceClassification* class so that the size and number of linear layers are automatically selected according to the model. We performed an optimization process of the model using *Optuna* [36], with 20 trials for hyperparameter search maximizing the macro averaged F1 score. The AdamW optimizer (β_1=0.9, β_2=0.999, ϵ=1e−8) was used with *learning rate* and number of epochs n searched in $\gamma \in [1e{-}7, 1e{-}3]$ and $n \in [1, 10]$, respectively.

For the *acoustic*-based baseline we replicated the method described in [6], hereafter referred to as S2B, an alignment process between BERT-based and speech embeddings. For this, we ran the pretraining using three different approaches to encode the speech: *(i)* 80-dimensional log-Mel Filter-Bank (LFB) features over 25 ms frames (10 ms rate) from the input speech signal; *(ii)* our inhouse XLSR-53 pretrained acoustic model without any adaptation to SLURP (✗); and, *(iii)* our XLSR-53 model adapted to SLURP dataset (✓). Finally, as a reference, we include the results reported by Villatoro et al. [15], where the original WCN-BERT method described in [16] was implemented and evaluated on SLURP. However, these results are not directly comparable as the WCN-BERT approach incorporates a different utterance representation which explicitly considers additional structural features at the moment of generating the `[CLS]` representation token.

3.3. Results

Table 1 shows the alignment accuracies (ACC) obtained by the pretrained models, S2B and WCN2B, on the People's Speech *dev* set. As described in §2.3, generated embeddings $\mathbf{B}^{\text{wcn}}$ are compared with BERT embeddings. Thus, the ACC value indicates how similar are $\mathbf{B}^{\text{wcn}}$ and $\mathbf{B}^{\text{text}}$ (Fig. 1). Notice that independently of the XLSR-53 adaptation to SLURP, the ACC of our WCN2B approach always outperforms the equivalent speech-to-text (S2B) alignment approach.

Table 2 shows the obtained IC results. Column "Model" depicts the model's configuration used for the corresponding experiment. Column "XLSR-53 adaptation" indicates whether or not the XLSR-53 model was fine-tuned to SLURP. Notably, *oracle* results obtained after the BERT-optimization process, represent a strong baseline (F1=0.88 and ACC=0.91) which outperforms, to the best of our knowledge, text-based results reported on SLURP, resulting in an additional contribution of the paper.[4] Among the results obtained by the S2B method, notice that traditional LFB features allow the speech-to-text alignment architecture to outperform results obtained by the same architecture when the pretrained not-adapted XLSR-53 speech encoder is used. However, if the adaptation step is applied, results get a 7.8% relative improvement from F1=0.64 to F1=0.69 on the *test* partition. Finally, notice that the proposed WCN-to-Text alignment method (WCN2B-adapted) yields a relative improvement of 17.1% and of 9.4% in F1 and ACC respectively, over the $S2B_{LFB}$ method, reaching an F1=0.75 and an ACC=0.81. Similarly, the proposed WCN2B relatively improves the ACC by 3.7% in comparison to the results reported in [15], even though our WCN2B approach does not explicitly add any additional structure information to the `[CLS]` token. As ablation experiments, we ran the pretraining of the proposed WCN2B method proportionally reducing the transformer layers and the attention heads from 12 to 2. Obtained results on the SLURP test set showed a drop in F1 from 0.75→0.72 and, in ACC from 0.81→0.79, indicating that a deeper model captures long-term dependencies in the WCN sequence more effectively, and that if computational power is limited, reducing the model size won't severely impact performance.

Table 2. Macro F1-score (F1) and Accuracy (ACC) on intent classification (IC) task for all evaluated models.

Model	XLSR-53 adaptation	*Dev* (↑) ACC	*Dev* (↑) F1	*Test* (↑) ACC	*Test* (↑) F1
Text-based (conventional pipeline SLU)					
Oracle	*NA*	*0.91*	*0.90*	*0.91*	*0.88*
1-*best*	✗	0.71	0.61	0.71	0.62
1-*best*	✓	0.84	0.81	0.85	0.80
Acoustic-based (E2E approach)					
$S2B_{(LFB)}$	✗	0.75	0.67	0.74	0.64
$S2B_{(XLSR53)}$	✗	0.73	0.66	0.72	0.62
$S2B_{(XLSR53)}$	✓	**0.80**	**0.75**	**0.79**	**0.69**
WCN-based					
$WCN2B_{(XLSR53)}$	✗	**0.70**	0.61	**0.69**	0.62
Villatoro et al. [15]*	✗	0.68	**0.67**	0.68	**0.68**
$WCN2B_{(XLSR53)}$	✓	**0.80**	0.72	**0.81**	**0.75**
Villatoro et al. [15]*	✓	0.78	**0.77**	0.79	0.79

4. CONCLUSIONS

In this work, we implemented a method for exploiting pretrained BERT as a teacher to inject fine-grained token-level embedding information into WCNs encodings. Using a recently proposed contrastive learning objective, our architecture is capable of learning WCN-to-text alignments at the bin level. An exhaustive evaluation on SLURP showed that the proposed architecture can outperform recent speech-to-BERT alignment methods by a 17.1% relative F1 improvement.

[4] At the moment of writing this paper, January 2024, the best accuracy result reported in the leaderboard for Intent Classification on SLURP was ACC=90%. (`https://paperswithcode.com/sota/intent-classification-on-slurp`)

5. REFERENCES

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

[2] Martin Fajcik, Muskaan Singh, Juan Pablo Zuluaga-Gomez, and et al., "IDIAPers @ causal news corpus 2022: Extracting cause-effect-signal triplets via pre-trained autoregressive language model," in *Proceedings of the 5th CASE Workshop@EMNLP*, 2022.

[3] Suman Ravuri and Andreas Stolcke, "Recurrent neural network and lstm models for lexical utterance classification," in *Proc. Interspeech*, 2015, pp. 135–139.

[4] Dmitriy Serdyuk, Yongqiang Wang, Christian Fuegen, Anuj Kumar, Baiyang Liu, and Yoshua Bengio, "Towards end-to-end spoken language understanding," in *Proc. ICASSP*, 2018, pp. 5754–5758.

[5] Bidisha Sharma, Maulik C. Madhavi, and Haizhou Li, "Leveraging acoustic and linguistic embeddings from pretrained speech and language models for intent classification," in *Proc. ICASSP*. IEEE, 2021.

[6] Vishal Sunder, Eric Fosler-Lussier, Samuel Thomas, Hong-Kwang J Kuo, and Brian Kingsbury, "Tokenwise contrastive pretraining for finer speech-to-bert alignment in end-to-end speech-to-intent systems," in *Proc. Interspeech*, 2022.

[7] Matthew Henderson, Milica Gašić, Blaise Thomson, Pirros Tsiakoulis, Kai Yu, and Steve Young, "Discriminative spoken language understanding using word confusion networks," in *2012 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2012, pp. 176–181.

[8] Su Zhu, Ouyu Lan, and Kai Yu, "Robust spoken language understanding with unsupervised asr-error adaptation," in *Proc. ICASSP*. IEEE, 2018, pp. 6179–6183.

[9] Jean-Philippe Robichaud, Paul A Crook, Puyang Xu, Omar Zia Khan, and Ruhi Sarikaya, "Hypotheses ranking for robust domain classification and tracking in dialogue systems," in *Proc. Interspeech*, 2014.

[10] Omar Zia Khan, Jean-Philippe Robichaud, Paul A Crook, and Ruhi Sarikaya, "Hypotheses ranking and state tracking for a multi-domain dialog system using multiple asr alternates," in *Proc. Interspeech*, 2015.

[11] Jan Švec, Luboš Šmídl, Tomáš Valenta, Adam Chỳlek, and Pavel Ircing, "Word-semantic lattices for spoken language understanding," in *Proc. ICASSP*. IEEE, 2015, pp. 5266–5270.

[12] Chao-Wei Huang and Yun-Nung Chen, "Adapting pretrained transformer to lattices for spoken language understanding," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2019, pp. 845–852.

[13] Dilek Hakkani-Tür, Frédéric Béchet, Giuseppe Riccardi, and Gokhan Tur, "Beyond asr 1-best: Using word confusion networks in spoken language understanding," *Computer Speech & Language*, vol. 20, no. 4, pp. 495–514, 2006.

[14] Esaú Villatoro-Tello, Srikanth Madikeri, Petr Motlicek, Aravind Ganapathiraju, and Alexei V. Ivanov, "Expanded lattice embeddings for spoken document retrieval on informal meetings," in *Proceedings of the 45th ACM SIGIR Conference*, 2022, SIGIR '22.

[15] Esaú Villatoro-Tello, Srikanth Madikeri, Juan Zuluaga-Gomez, and Et al., "Effectiveness of text, acoustic, and lattice-based representations in spoken language understanding tasks," in *Proc. ICASSP*, 2023.

[16] Chen Liu, Su Zhu, Zijian Zhao, Ruisheng Cao, Lu Chen, and Kai Yu, "Jointly encoding word confusion network and dialogue context with bert for spoken language understanding," in *Proc. Interspeech*, 2020.

[17] 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, pp. 814–818.

[18] Lasse Borgholt, Jakob Drachmann Havtorn, Mostafa Abdou, Joakim Edin, Lars Maaløe, Anders Søgaard, and Christian Igel, "Do we still need automatic speech recognition for spoken language understanding?," *arXiv preprint arXiv:2111.14842*, 2021.

[19] Yidi Jiang, Bidisha Sharma, Maulik Madhavi, and Haizhou Li, "Knowledge Distillation from BERT Transformer to Speech Transformer for Intent Classification," in *Proc. Interspeech*, 2021, pp. 4713–4717.

[20] Bidisha Sharma, Maulik Madhavi, Xuehao Zhou, and Haizhou Li, "Exploring teacher-student learning approach for multi-lingual speech-to-intent classification," in *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2021, pp. 419–426.

[21] Lidia Mangu, Eric Brill, and Andreas Stolcke, "Finding consensus in speech recognition: word error minimization and other applications of confusion networks," *Computer Speech & Language*, vol. 14, no. 4, pp. 373–400, 2000.

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

[23] Daniel Galvez, Greg Diamos, Juan Manuel Ciro Torres, and Et al., "The people's speech: A large-scale diverse english speech recognition dataset for commercial usage," in *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021.

[24] Emanuele Bastianelli, Andrea Vanzo, et al., "SLURP: A spoken language understanding resource package," in *Proc. of the 2020 EMNLP*, 2020, pp. 7252–7262.

[25] Alexis Conneau, Alexei Baevski, Ronan Collobert, Abdelrahman Mohamed, and Michael Auli, "Unsupervised cross-lingual representation learning for speech recognition," *arXiv preprint arXiv:2006.13979*, 2020.

[26] Jean Carletta, Simone Ashby, et al., "The ami meeting corpus: A pre-announcement," in *International workshop on machine learning for multimodal interaction*. Springer, 2005, pp. 28–39.

[27] John J Godfrey, Edward C Holliman, and Jane McDaniel, "Switchboard: Telephone speech corpus for research and development," in *Acoustics, speech, and signal processing, ieee international conference on*. IEEE Computer Society, 1992, vol. 1, pp. 517–520.

[28] Yiwen Shao, Yiming Wang, Daniel Povey, and Sanjeev Khudanpur, "Pychain: A fully parallelized pytorch implementation of lf-mmi for end-to-end asr," *arXiv preprint arXiv:2005.09824*, 2020.

[29] Hossein Hadian, Hossein Sameti, Daniel Povey, and Sanjeev Khudanpur, "End-to-end speech recognition using lattice-free mmi.," in *Proc. Interspeech*, 2018, pp. 12–16.

[30] Apoorv Vyas, Srikanth Madikeri, and Hervé Bourlard, "Comparing CTC and LFMMI for out-of-domain adaptation of wav2vec 2.0 acoustic model," in *Proc. Interspeech*, 2021.

[31] Srikanth Madikeri et al., "Pkwrap: a pytorch package for lf-mmi training of acoustic models," *arXiv preprint arXiv:2010.03466*, 2020.

[32] Yiming Wang et al., "Espresso: A fast end-to-end neural speech recognition toolkit," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2019.

[33] Banriskhem Khonglah et al., "Incremental semi-supervised learning for multi-genre speech recognition," in *Proc. ICASSP*, 2020.

[34] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al., "The kaldi speech recognition toolkit," in *2011 IEEE ASRU Workshop*. IEEE Signal Processing Society, 2011.

[35] Haihua Xu, Daniel Povey, Lidia Mangu, and Jie Zhu, "Minimum bayes risk decoding and system combination based on a recursion for edit distance," *Computer Speech & Language*, vol. 25, no. 4, pp. 802–828, 2011.

[36] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama, "Optuna: A next-generation hyperparameter optimization framework," in *Proc. of the 25th ACM SIGKDD*, 2019.