N-Best ASR Transformer: Enhancing SLU Performance using Multiple ASR Hypotheses

Karthik Ganesan*, **Pakhi Bamdev***, **Jaivarsan B***, **Amresh Venugopal, Abhinav Tushar**
Vernacular.ai
{karthikganesan17, pakhi.bamdev.in}@gmail.com
{jaivarsan, amresh, abhinav}@vernacular.ai

Abstract

Spoken Language Understanding (SLU) systems parse speech into semantic structures like dialog acts and slots. This involves the use of an Automatic Speech Recognizer (ASR) to transcribe speech into multiple text alternatives (hypotheses). Transcription errors, common in ASRs, impact downstream SLU performance negatively. Approaches to mitigate such errors involve using richer information from the ASR, either in form of N-best hypotheses or word-lattices. We hypothesize that transformer models learn better with a simpler utterance representation using the concatenation of the N-best ASR alternatives, where each alternative is separated by a special delimiter [SEP]. In our work, we test our hypothesis by using concatenated N-best ASR alternatives as the input to transformer encoder models, namely BERT and XLM-RoBERTa, and achieve performance equivalent to the prior state-of-the-art model on DSTC2 dataset. We also show that our approach significantly outperforms the prior state-of-the-art when subjected to the low data regime. Additionally, this methodology is accessible to users of third-party ASR APIs which do not provide word-lattice information.

1 Introduction

Spoken Language Understanding (SLU) systems are an integral part of Spoken Dialog Systems. They parse spoken utterances into corresponding semantic structures e.g. dialog acts. For this, a spoken utterance is usually first transcribed into text via an Automated Speech Recognition (ASR) module. Often these ASR transcriptions are noisy and erroneous. This can heavily impact the performance of downstream tasks performed by the SLU systems.

To counter the effects of ASR errors, SLU systems can utilise additional feature inputs from ASR. A common approach is to use N-best hypotheses where multiple ranked ASR hypotheses are used, instead of only 1 ASR hypothesis. A few ASR systems also provide additional information like word-lattices and word confusion networks. Word-lattice information represents alternative word-sequences that are likely for a particular utterance, while word confusion networks are an alternative topology for representing a lattice where the lattice has been transformed into a linear graph. Additionally, dialog context can help in resolving ambiguities in parses and reducing impact of ASR noise.

N-best hypotheses: Li et al. (2019) work with 1-best ASR hypothesis and exploits unsupervised ASR error adaption method to map ASR hypotheses and transcripts to a similar feature space. On the other hand, Khan et al. (2015) uses multiple ASR hypotheses to predict multiple semantic frames per ASR choice and determine the true spoken dialog system's output using additional context. **Word-lattices:** Ladhak et al. (2016) propose using recurrent neural networks (RNNs) to process weighted lattices as input to SLU. Švec et al. (2015) presents a method for converting word-based ASR lattices into word-semantic (W-SE) which reduces the sparsity of the training data. Huang and Chen (2019) provides an approach for adapting lattices with pretrained transformers. **Word confusion networks (WCN):** Jagfeld and Vu (2017) proposes a technique to exploit word confusion networks (WCNs) as training or testing units for slot filling. Masumura et al. (2018) models WCN as sequence of bag-of-weighted-arcs and introduce a mechanism that converts the bag-of-weighted-arcs into a continuous representation to build a neural network based spoken utterance classification. Liu et al. (2020) proposes a BERT based SLU model to encode WCNs and the dialog context jointly to

* The first three authors have equal contribution.

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Short Papers), pages 93–98
August 1–6, 2021.

reduce ambiguity from ASR errors and improve SLU performance with pre-trained models.

The motivation of this paper is to improve performance on downstream SLU tasks by exploiting *transfer learning* capabilities of the pre-trained transformer models. Richer information representations like word-lattices (Huang and Chen (2019)) and word confusion networks (Liu et al. (2020)) have been used with GPT and BERT respectively. These representations are non-native to Transformer models, that are pre-trained on plain text sequences. We hypothesize that transformer models will learn better with a simpler utterance representation using concatenation of the N-best ASR hypotheses, where each hypothesis is separated by a special delimiter [SEP]. We test the effectiveness of our approach on a dialog state tracking dataset - DSTC2 (Henderson et al., 2014), which is a standard benchmark for SLU.

Contributions: (i) Our proposed approach, trained with a simple input representation, exceeds the competitive baselines in terms of accuracy and shows equivalent performance on the F1-score to the prior state-of-the-art model. (ii) We significantly outperform the prior state-of-the-art model in the low data regime. We attribute this to the effective *transfer learning* from the pre-trained Transformer model. (iii) This approach is accessible to users of third party ASR APIs unlike the methods that use word-lattices and word confusion networks which need deeper access to the ASR system.

2 N-Best ASR Transformer

N-Best ASR Transformer[1] works with a simple input representation achieved by concatenating the N-Best ASR hypotheses together with the dialog context (system utterance). Pre-trained transformer models, specifically BERT and XLMRoBERTa, are used to encode the input representation. For output layer, we use a semantic tuple classifier (STC) to predict *act-slot-value* triplets. The following sub-sections describe our approach in detail.

2.1 Input Representation

For representing the input we concatenate the last system utterance S (dialog context), and the user utterance U. U is represented as concatenation of the N-best[2] ASR hypotheses, separated by a special delimiter, [SEP]. The final representation is shown in equation 1 below:

$$x_i = [\text{CLS}] \oplus \text{TOK}(S_i) \oplus \bigoplus_{j=1}^{N} (\text{TOK}(U_i^j) \oplus [\text{SEP}]) \quad (1)$$

Here, U_i^j refers to the j^{th} ASR hypothesis for the i^{th} sample, $\oplus$ denotes the concatenation operator, TOK(.) is the tokenizer, [CLS] and [SEP] are the special tokens.

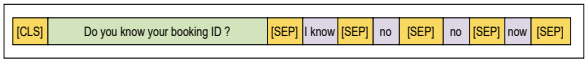

Figure 1: Input representation: The green boxes represents the last system utterances followed by ASR hypotheses of user utterances concatenated together with a [SEP] token.

As represented in figure 2, we also pass segment IDs along with the input to differentiate between segment a (last system utterance) and segment b (user utterance).

2.2 Transformer Encoder

The above mentioned input representation can be easily used with any pre-trained transformer model. For our experiments, we select BERT (Devlin et al., 2019) and XLM-RoBERTa[3] (Conneau et al., 2020) for their recent popularity in NLP research community.

2.3 Output Representation

The final hidden state of the transformer encoder corresponding to the special classification token [CLS] is used as an aggregated input representation for the downstream classification task by a semantic tuple classifier (STC) (Mairesse et al., 2009). STC uses two classifiers to predict the *act-slot-value* for a user utterance. A binary classifier is used to predict the presence of each *act-slot* pair, and a multi-class classifier is used to predict the *value* corresponding to the predicted act-slot pairs. We omit the latter classifier for the act-slot pairs with no value (like *goodbye*, *thankyou*, *request_food* etc.).

[1]The code is available at https://github.com/Vernacular-ai/N-Best-ASR-Transformer

[2]We use ASR transcriptions (N $\leq$ 10) provided by DSTC2 dataset to perform our experiments. Our input structure can support variable N during training and inference.

[3]The model name XLM-RoBERTa and XLM-R will be used interchangeably throughout the paper.

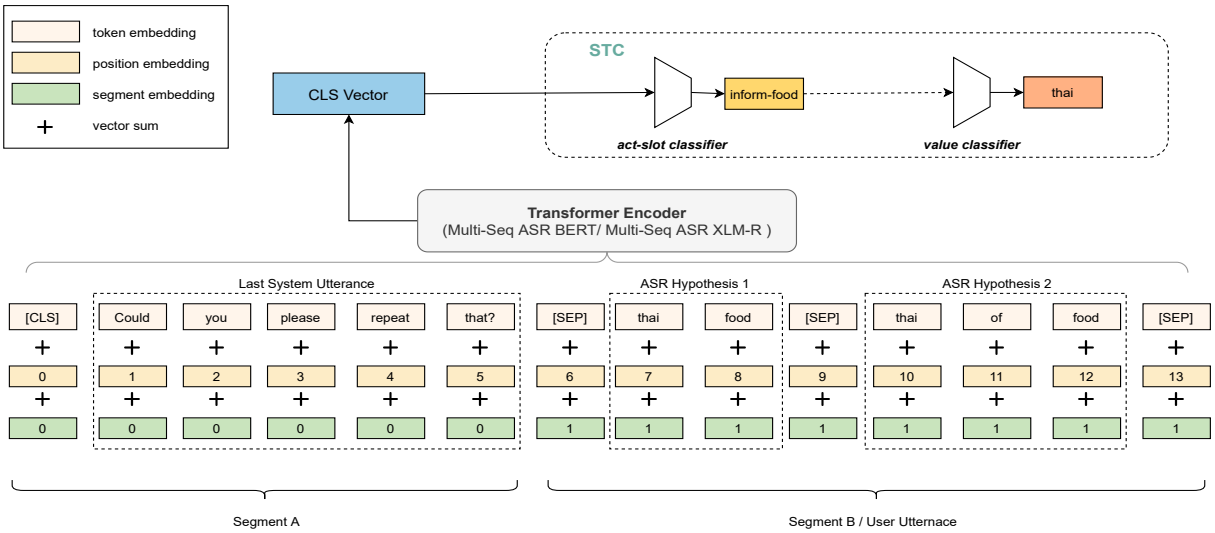

Figure 2: **N-Best ASR Transformer:** The input representation is encoded by a transformer model which forms an input for a Semantic Tuple Classifier (STC). STC uses binary classifiers to predict the presence of act-slot pairs, followed by a multi-class classifier that predicts the value for each act-slot pair.

3 Experimental Setup

3.1 Dataset

We perform our experiments on data released by the Dialog State Tracking Challenge (DSTC2) (Henderson et al., 2014). It includes pairs of utterances and the corresponding set of *act-slot-value* triplets for training (11,677 samples), development (3,934 samples), and testing (9,890 samples). The task in the dataset is to parse the user utterances like "*I want a moderately priced restaurant.*" into a corresponding semantic representation in the form of "*inform(pricerange=moderate)*" triplet. For each utterance, both the manual transcription and a maximum of 10-best ASR hypotheses are provided. The utterances are annotated with multiple *act-slot-value* triplets. For transcribing the utterances DSTC2 uses two ASRs - one with an artificially degraded statistical acoustic model, and one which is fully optimized for the domain. Training and development sets include transcriptions from both the ASRs. To utilise this dataset we first transform it into the input format as discussed in section 2.1.

3.2 Baselines

We compare our approach with the following baselines:

- **SLU2** (Williams, 2014): Two binary classifiers (decision trees) are used with word n-grams from the ASR N-best list and the word confusion network. One predicts the presence of that slot-value pair in the utterance and the other estimate for each user dialog act.

- **CNN+LSTMw4** (Rojas-Barahona et al., 2016): A convolution neural network (CNN) is trained with the N-best ASR hypotheses to output the utterance representation. A long-short term memory network (LSTM) with a context window size of 4 outputs a context representation. The models are jointly trained to predict for the act-slot pair. Another model with the same architecture is trained to predict for the value corresponding to the predicted act-slot pair.

- **CNN** (Zhao and Feng, 2018): Proposes CNN based models for dialog act and slot-type prediction using 1-best ASR hypothesis.

- **Hierarchical Decoding** (Zhao et al., 2019): A neural-network based binary classifier is used to predict the act and slot type. A hybrid of sequence-to-sequence model with attention and pointer network is used to predict the value corresponding to the detected act-slot pair.1-Best ASR hypothesis was used for both training and evaluation tasks.

- **WCN-BERT + STC** (Liu et al., 2020): Input utterance is encoded using the Word Confusion Network (WCN) using BERT by having the same position ids for all words in the bin of a lattice and modifying self-attention to work with word probabilities. A semantic tuple classifier uses a binary classifier to predict the act-slot value, followed by a multi-class classifier that predicts the value corresponding

to the act-slot tuple.

3.3 Experimental Settings

We perform hyper-parameter tuning on the validation set to get optimal values for dropout rate δ, learning rate lr, and the batch size b. Based on the best F1-score, the final selected parameters were δ = 0.3, lr = 3e-5 and b = 16. We set the warm-up rate wr = 0.1, and L2 weight decay $L2$ = 0.01. We make use of Huggingface's *Transformers* library (Wolf et al., 2020) to fine-tune the *bert-base-uncased* and *xlm-roberta-base*, which is optimized over Huggingface's BertAdam optimizer. We trained the model on Nvidia T4 single GPU on AWS EC2 g4dn.2xlarge instance for 50 epochs. We apply early stopping and save the best-performing model based on its performance on the validation set.

4 Results

In this section, we compare the performance of our approach with the baselines on the DSTC2 dataset. To compare the *transfer learning* effectiveness of pre-trained transformers with *N-Best ASR BERT* (our approach) and the previous state-of-the-art model *WCN-BERT STC*, we perform comparative analysis in the low data regime. Additionally, we perform an ablation study on *N-Best ASR BERT* to see the impact of modeling dialog context (last system utterance) with the user utterances.

4.1 Performance Evaluation

Model	F1-score	Accuracy
SLU2	82.1	-
CNN+LSTM_w4	83.6	-
CNN	85.3	-
Hierarchical Decoding	86.9	-
WCN-BERT + STC	**87.9**	81.1
N-Best ASR XLM-R (Ours)	87.4	**81.9**
N-Best ASR BERT (Ours)	87.8	81.8

Table 1: F1-scores (%) and utterance-level accuracy (%) of baseline models and our proposed model on the test set.

Since the task is a multi-label classification of *act-slot-value* triplets, we report utterance level accuracy and F1-score. A prediction is correct if the set of labels predicted for a sample exactly matches the corresponding set of labels in the ground truth. As shown in Table 1, we compare our models, *N-Best ASR BERT* and *N-Best ASR XLM-R*, with baselines mentioned in section . Both of our proposed models, trained with concatenated N-Best ASR hypotheses, outperform the competitive baselines in terms of accuracy and show comparable performance on F1-score with *WCN-BERT STC*.

4.2 Performance in Low Data Regime

Train Data (%age)	WCN-BERT STC	*N-Best ASR BERT*
5	78.5	**83.9**
10	80.3	**85.5**
20	84.4	**86.7**
50	85.9	**87.7**

Table 2: F1-scores (%) for our proposed model *N-Best ASR BERT* (ours) and *WCN-BERT STC* (previous state-of-the-art.

To study the performance of model in the low data regime, we randomly select p percentage of samples from the training set in a stratified fashion, where $p \in \{5, 10, 20, 50\}$. We pick our model *N-Best ASR BERT* and *WCN-BERT STC* for this study because both use BERT as the encoder model. For both models, we perform experiments using the same training, development, and testing splits. From Table 2, we find that *N-Best ASR BERT* outperforms *WCN-BERT STC* model significantly for low data regime, especially when trained on 5% and 10% of the training data. It shows that our approach effectively *transfer learns* from pre-trained transformer's knowledge. We believe this is due to the structural similarity between our input representation and the input BERT was pre-trained on.

4.3 Significance of Dialog Context

Model	Variation	F1-score	Accuracy
N-Best ASR BERT	without system utterance	86.5	80.2
	with system utterance	**87.8**	**81.8**

Table 3: F1-scores (%) and utterance-level accuracy (%) of our model *N-Best ASR BERT* on the test set when trained with and without system utterances.

Through this ablation study, we try to understand the impact of dialog context on model's performance. For this, we train *N-Best ASR BERT* in the following two settings:

- When input representation consists of only the user utterance.
- When input representation consists of both the last system utterance (dialog context) and the user utterance as shown in figure 3.

As presented in Table 3, we observe that modeling the last system utterance helps in achieving better F1 and utterance-level accuracy by the difference of 1.3% and 1.6% respectively.

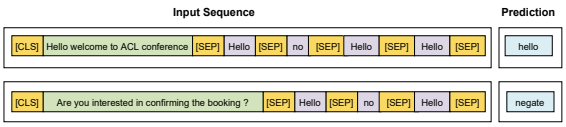

Figure 3: Significance of Dialog Context: The green box depicts the dialog context that helps disambiguate the very similar ASR hypotheses shown in purple boxes.

It proves that dialog context helps in improving the performance of downstream SLU tasks. Figure 3 represents one such example where having dialog context in form of the last system utterance helps disambiguate between the two similar user utterances.

5 Conclusion

In this work, building on a simple input representation, we propose *N-Best ASR Transformer*, which outperforms all the competitive baselines on utterance-level accuracy for the DSTC2 dataset. However, the highlight of our work is in achieving significantly higher performance in an extremely low data regime. This approach is accessible to users of third-party ASR APIs, unlike the methods that use word-lattices and word confusion networks. As future extensions to this work, we plan to :

- Enable our proposed model to generalize to out-of-vocabulary (OOV) slot values.
- Evaluate our approach in a multi-lingual setting.
- Evaluate on different values N in N-best ASR.
- Compare the performance of our approach on ASRs with different Word Error Rates (WERs).

Acknowledgement

We are highly grateful to our organization Vernacular.ai and our Machine Learning Team for (i) exposing us to practical problems related to multilingual voice-bots, (ii) giving us access to resources to solve this problem, (iii) helping us deploy this work in production for real-world users, and (iv) for their excellent feedback on this work.

References

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Matthew Henderson, Blaise Thomson, and Jason D Williams. 2014. The second dialog state tracking challenge. In *Proceedings of the 15th annual meeting of the special interest group on discourse and dialogue (SIGDIAL)*, pages 263–272.

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

Glorianna Jagfeld and Ngoc Thang Vu. 2017. Encoding word confusion networks with recurrent neural networks for dialog state tracking. In *Proceedings of the Workshop on Speech-Centric Natural Language Processing*, pages 10–17, Copenhagen, Denmark. Association for Computational Linguistics.

Omar Zia Khan, Jean-Philippe Robichaud, Paul A Crook, and Ruhi Sarikaya. 2015. Hypotheses ranking and state tracking for a multi-domain dialog system using multiple asr alternates. In *Sixteenth Annual Conference of the International Speech Communication Association*.

Faisal Ladhak, Ankur Gandhe, Markus Dreyer, Lambert Mathias, Ariya Rastrow, and Björn Hoffmeister. 2016. Latticernn: Recurrent neural networks over lattices. In *Interspeech*, pages 695–699.

Hao Li, Chen Liu, Su Zhu, and Kai Yu. 2019. Robust spoken language understanding with acoustic and domain knowledge. In *2019 International Conference on Multimodal Interaction*, pages 531–535.

Chen Liu, Su Zhu, Zijian Zhao, Ruisheng Cao, Lu Chen, and Kai Yu. 2020. Jointly encoding word confusion network and dialogue context with BERT for spoken language understanding. In *Interspeech 2020, 21st Annual Conference of the International Speech Communication Association, Virtual Event, Shanghai, China, 25-29 October 2020*, pages 871–875. ISCA.

François Mairesse, Milica Gasic, Filip Jurcicek, Simon Keizer, Blaise Thomson, Kai Yu, and Steve Young. 2009. Spoken language understanding from unaligned data using discriminative classification models. In *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4749–4752. IEEE.

Ryo Masumura, Yusuke Ijima, Taichi Asami, Hirokazu Masataki, and Ryuichiro Higashinaka. 2018. Neural confnet classification: Fully neural network based spoken utterance classification using word confusion networks. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6039–6043. IEEE.

Lina M. Rojas-Barahona, Milica Gašić, Nikola Mrkšić, Pei-Hao Su, Stefan Ultes, Tsung-Hsien Wen, and Steve Young. 2016. Exploiting sentence and context representations in deep neural models for spoken language understanding. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 258–267, Osaka, Japan. The COLING 2016 Organizing Committee.

Jan Švec, Luboš Šmídl, Tomáš Valenta, Adam Chỳlek, and Pavel Ircing. 2015. Word-semantic lattices for spoken language understanding. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5266–5270. IEEE.

Jason D Williams. 2014. Web-style ranking and slu combination for dialog state tracking. In *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 282–291.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Lin Zhao and Zhe Feng. 2018. Improving slot filling in spoken language understanding with joint pointer and attention. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 426–431.

Zijian Zhao, Su Zhu, and Kai Yu. 2019. A hierarchical decoding model for spoken language understanding from unaligned data. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7305–7309. IEEE.