TokenVerse: Towards Unifying Speech and NLP Tasks via Transducer-based ASR

Shashi Kumar[1,2] **Srikanth Madikeri**[1,3] **Juan Zuluaga-Gomez**[1] **Iuliia Thorbecke**[1,3]
Esaú Villatoro-Tello[1] **Sergio Burdisso**[1] **Petr Motlicek**[1,4]
Karthik Pandia[5] **Aravind Ganapathiraju**[5]
[1]Idiap Research Institute, Switzerland; [2]EPFL, Switzerland; [3]University of Zurich, Switzerland;
[4]Brno University of Technology, Czech Republic; [5]Uniphore, India
shashi.kumar@idiap.ch

Abstract

In traditional conversational intelligence from speech, a cascaded pipeline is used, involving tasks such as voice activity detection, diarization, transcription, and subsequent processing with different NLP models for tasks like semantic endpointing and named entity recognition (NER). Our paper introduces TokenVerse, a single Transducer-based model designed to handle multiple tasks. This is achieved by integrating task-specific tokens into the reference text during ASR model training, streamlining the inference and eliminating the need for separate NLP models. In addition to ASR, we conduct experiments on 3 different tasks: speaker change detection, endpointing, and NER. Our experiments on a public and a private dataset show that the proposed method improves ASR by up to 7.7% in relative WER while outperforming the cascaded pipeline approach in individual task performance. Our code is publicly available: https://github.com/idiap/tokenverse-unifying-speech-nlp

1 Introduction

Automated analysis of conversational audios has a wide range of practical applications, including in contact center analytics (Saberi et al., 2017; Mamou et al., 2006). Traditionally, conversational audios are transcribed with intermediate voice activity detection (VAD) (Medennikov et al., 2020) or endpointing (Chang et al., 2019) and diarization (Park et al., 2022). Afterward, separate NLP pipelines are employed on the transcripts to perform tasks such as named entity recognition (NER) (Li et al., 2020), among others, to comprehend the conversation's structure and content (Zou et al., 2021; Xu et al., 2021). Using separate models for each subtask (optimized independently) has drawbacks (Ghannay et al., 2018) such as error propagation and a potential mismatch between automatic speech recognition (ASR) metrics and the final task. For instance, the best ASR hypothesis may not be optimal for the final task. Moreover, the cascaded approaches could translate to increased compute and latency, which will be exacerbated by the introduction of a new task.

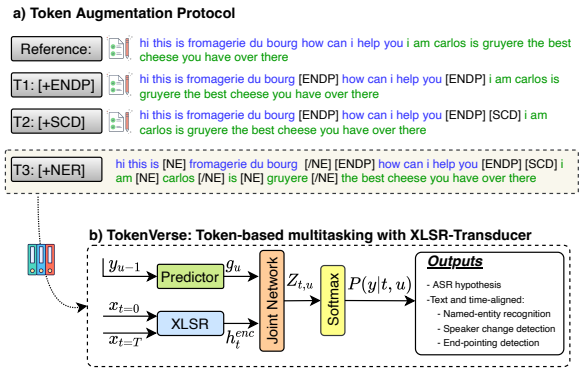

Figure 1: a) Proposed unified token augmentation protocol for SCD, ENDP, and NER. b) TokenVerse unifies multiple speech and NLP tasks (e.g., *T1+T2+T3*) in a single model within the neural Transducer framework.

In this paper, we introduce TokenVerse, a neural Transducer (Graves, 2012) model capable of learning ASR and multiple additional tasks through the incorporation of task tokens. In contrast to the multi-head based multitasking approaches explored in previous studies (Chen et al., 2021; wen Yang et al., 2021; Kumar et al., 2024), TokenVerse distinguishes itself by generating tokens directly within the ASR hypothesis, as illustrated in Fig. 1a. Leveraging the transducer architecture (Graves, 2012), we can attain text-audio alignment for each output token, including those designated as task tokens. For example, we can perform NER directly in the acoustic domain, presenting potential utility in scenarios such as audio de-identification (Cohn et al., 2019). To address challenges in low-resource settings, we use self-supervised (SSL) trained XLSR-53 (Conneau et al., 2020) model as an encoder in the transducer setup, leading to the XLSR-Transducer (Fig. 1b). Previous works aims at modeling several tasks directly from speech using special tokens (Wu et al., 2024; Chang et al.,

2023), or ASR with speaker change detection (SCD) (Shafey et al., 2019; Xia et al., 2022; Kumar et al., 2024), VAD (Radford et al., 2023), speech-to-text translation (Zuluaga-Gomez et al., 2023), or timestamps (Cornell et al., 2023), NER (Ghannay et al., 2018; Yadav et al., 2020) and multi-speaker ASR (Kanda et al., 2022; Wu et al., 2023). Token-based multitasking offers multiple benefits, e.g., it has a fix number of parameters while all tasks are predicted with standard decoding without increased latency. However, NLP tasks like NER in conjunction with other tasks from audio domains have not received much attention in the literature. Therefore, we consider 3 additional tasks alongside ASR: SCD, endpointing and NER. These tasks are selected to represent both audio and NLP domains. SCD is an audio task (Bredin et al., 2017). Endpointing can be viewed as an NLP task when conducting semantic endpointing (Raux and Eskenazi, 2008), or as an audio task (Chang et al., 2019). NER is an NLP task (Li et al., 2020; Ghannay et al., 2018). They serve as suitable benchmarks for evaluating our proposed method.

2 TokenVerse

Through `TokenVerse`, we aim to train a single model for ASR (main task), speaker change detection (SCD), endpointing, and named entity recognition (NER). This is achieved by augmenting the reference text, with task tokens that denote special events at the acoustic level.

2.1 Token Augmentation Protocol

We introduce “tokens" for tasks apart from ASR: `[SCD]` (speaker change detection), `[NE]` and `[/NE]` (named entity recognition), and `[ENDP]` (endpointing) to prepare the multitask dataset. An illustrative example is depicted in Figure 1a. We insert `[SCD]` token during text concatenation if there is a speaker change within an utterance. The `[ENDP]` token is inserted at the end of a segment text, considered as a semantic endpoint from the conversational context perspective. Note that occurrence of `[ENDP]` will be a superset of `[SCD]` because a speaker change indicates the completion of the previous speaker's sentence. For NER, we insert `[NE]` before the start of a named entity and `[/NE]` after it is concluded, since it can comprise multiple words.

2.2 Training & Inference

`TokenVerse` Training We train the XLSR-Transducer model on the multitask data which consists of XLSR encoder, state-less predictor (Ghodsi et al., 2020) and joint networks (linear layer). The model is trained with pruned transducer loss (Kuang et al., 2022). We utilize SentencePiece (Kudo and Richardson, 2018) tokenizer to train subwords from the training text (Sennrich et al., 2016). Note that the text includes task-specific tokens, and splitting them into multiple subwords may degrade their prediction accuracy because the entire sequence of subwords for a token must be predicted correctly to count it as a valid token prediction. Hence, we ensure that tokens are represented by a single subword during their training.[1]

`TokenVerse` Inference We generate hypothesis with beam search. From the hypothesis, we can extract and align the predicted task tokens in the time domain. Since NER consists of two tokens, we extract words between a matched pairs of `[NE]` and `[/NE]`. To obtain timestamps for `[SCD]` or `[ENDP]`, we note the acoustic frame index for which these tokens are emitted and calculate time information, i.e., XLSR acoustic embeddings have a frame duration of 25ms and a stride of 20ms. Particularly for `[SCD]`, the time-level token prediction enables subsequent tasks, e.g., diarization (Xia et al., 2022).

2.3 Ablations within TokenVerse

We conduct ablation experiments to understand how including or excluding tasks affects other tasks in the `TokenVerse`. Note that ASR is our primary task and is always included.

Single task For each task, we retain only the tokens specific to that task in the multitask dataset and train our ASR model. This eliminates any detractor tasks that may affect the task being evaluated and serves as a baseline in this paper.

Leave-one-task-out We exclude tokens of a single task from the multitask data and train our ASR model. This provides insights whether we should retain or discard any task in `TokenVerse` for optimal performance on a given task.

Task-Transfer Learning In multi-head multi-task architectures (Chen et al., 2021), a new task can be learnt by fine-tuning the model on the new task while keeping the base encoder and other heads frozen. We explore this for `TokenVerse` by fine-tuning the model, derived from the removal of a task, on the removed task. Furthermore, we evaluate its impact on both existing tasks and the performance of the new task in comparison to the

[1] https://github.com/google/sentencepiece

overall performance when all-tasks are included.

3 Task-Specific Baselines, Metrics & Evaluation Protocol

In this section, we describe strong independent baselines for each task considered in this work.

Automatic Speech Recognition We train our XLSR-Transducer model after removing all task tokens from the multitask dataset. This serves as a baseline for the ASR task. **Evaluation** It is evaluated with WER. For `TokenVerse` models, we remove task tokens from both the reference and hypothesis to compute WER for a fair comparison.

Named-Entity Recognition We finetune pretrained BERT[2] (Devlin et al., 2019) model on our datasets for subword-level NER classification, a commonly used approach for this task in the literature. We evaluate the models on both reference and hypothesis from the ASR model. **Evaluation** NER systems are usually evaluated by comparing their outputs against human annotations, either using an exact-match or soft-match approach (Li et al., 2020). We adapted these metrics to a scenario where the text comes from an ASR system. Detailed description in appendix A.

Speaker Change Detection We utilize the diarization pipeline[3] from PyAnnote (Bredin, 2023), which achieves state-of-the-art results (Plaquet and Bredin, 2023) across multiple datasets, to extract speaker change timestamps from the audio. In literature, the SCD is predominantly regarded as a task within the audio domain (Bredin et al., 2017), we opt not to establish an independent text-based baseline for this task. **Evaluation** We evaluate SCD in two ways: text-based (only valid for `TokenVerse`) and time-based. For both methods, predictions from `TokenVerse` are compared with the reference, and the F1 score is calculated. Detailed description in appendix A.

Endpointing Considering semantic endpointing, we fine-tune BERT (Devlin et al., 2019) for `[ENDP]` token classification on the multitask training text, termed as BERT-ENDP. Results are reported on both reference text and hypothesis text obtained from `TokenVerse`. From the audio perspective, we use segmentation pipeline[4] from PyAnnote to obtain endpoint timestamps. **Evaluation** It follows the same approach as for SCD. We also report false alarms (FA), missed speech (MS), and detection error rate (DER), which are common metrics in endpointing literature (Medennikov et al., 2020).

[2] https://huggingface.co/google-bert/bert-base-uncased

[3] https://huggingface.co/pyannote/speaker-diarization-3.1

[4] huggingface.co/pyannote/segmentation-3.0

Table 1: Datasets statistics with token metadata per subset for the public and private datasets.

Datasets metadata			Token-based metadata [%]				
subset	#utt/word	dur [h]	[SCD]	[NE]	[ENDP]	#NE	#uniq
DefinedAI dataset							
train	10k/359k	40	1.9	3.6	2.1	6.5k	2350
dev	559/20k	2.25	2.0	3.6	2.1	379	232
test	1.1k/42k	4.5	1.9	3.4	2.0	727	378
CallHome dataset							
train	2.7k/198k	13	6.3	2.9	8.7	2.8k	1414
dev	641/52k	3	7.2	3.0	10.4	779	466
test	339/23k	1.5	6.0	3.0	9.6	351	220

4 Experimental Setup

4.1 Datasets Descriptions

To train `TokenVerse`, we require conversational audio data with corresponding transcripts, NER, segment timestamps, and speaker annotations. We could not find a large-scale public dataset satisfying all the tasks. Thus, we opt for a private dataset , *DefinedAI*[5]. We also train and evaluate on the open-source *CallHome* English dataset.

DefinedAI contains stereo-audio/transcript pairs for contact center conversations between agents and customers. We upsampled audio from 8 kHz to 16 kHz to align with the XLSR-53 model's requirements. Each segment includes transcripts, speaker ID and NE annotations, facilitating multitask dataset preparation. This dataset spans health, banking and finance domains, which makes it particularly challenging due to variations in NEs.

CallHome English dataset (LDC97S42) contains natural conversational stereo-audios between multiple speakers. The transcript includes named entities annotation.This dataset poses challenges due to its natural conversational nature, known to be challenging for ASR modeling, and a large number of short segments without entities, differing from the DefinedAI dataset. Further details about these datasets are provided in Table 1.

4.2 Multitask Dataset Preparation

Our work is focused on conversational audios which is typically long in duration (avg 5 minutes)

[5] https://www.defined.ai/

Table 2: WER (%) for ASR on DefinedAI with TokenVerse. The task tokens are removed from both the reference and hypothesis for WER calculation.

Exp	Model	WER (↓)
1)	ASR (baseline)	15.3
2)	all-tasks	**14.7**
3-a)	single-[SCD]	15.1
3-b)	single-[NE]	**14.7**
3-c)	single-[ENDP]	**14.7**

Table 3: Text-based performances on the the [NE] (exact- and soft-match) and [ENDP]. P: precision; R: recall. †upper-bound: BERT model evaluated on text references. ‡model trained on [ENDP] or [NE] task.

Exp	Model	[NE]-Exact			[NE]-Soft			[ENDP]
		@P	@R	@F1	@P	@R	@F1	@F1
BERT: fine-tuned on DefinedAI								
b-1)	Eval. on Ref.†	80.0	77.0	78.5	91.6	87.9	89.7	81.6
b-2)	Eval on Hyp.	52.9	53.0	52.9	82.0	81.3	81.6	80.5
2)	all-tasks	65.0	51.7	**57.6**	93.0	73.2	**81.9**	**89.9**
3-b/c)	single‡	61.7	49.9	55.2	91.4	73.3	81.4	88.5

and can't be directly used for ASR training due to high GPU memory requirements. The dataset provides audio-text transcripts together with timestamp information for every segment within the long-form audio. For each sample, we begin with the first segment $start$ and find the farthest segment end such that the duration is up to 20 seconds. Audios within this range are extracted as one utterance and this procedure is repeated until the last segment is consumed. Note that an utterance may span over multiple segments, potentially containing silences, noise, speaker changes, endpoints and numerous named entities. Afterward, we concatenate the text corresponding to all segments within an utterance, inserting token at appropriate positions according to our tasks, described in §2.1. This multitask dataset preparation approach applies universally across all datasets used in our experiments.

4.3 Training Details

We train TokenVerse on the multitask dataset. We implement the XLSR-Transducer model from the Icefall's Transducer recipe[6] adapted with XLSR from fairseq (Ott et al., 2019). The model is optimized with pruned RNN-t loss (Kuang et al., 2022). The initial learning rate is set to $lr = 1.25e^{-3}$ and we train the model for 50 epochs. For each dataset, the best epoch is selected based on WER on respective dev sets and results are presented on the eval sets. The task-transfer experiments (see §2.3) are trained for additional 10 epochs on the new task.

5 Results & Discussion

Automatic Speech Recognition For the *DefinedAI* (Tab. 2) set, including all tasks in TokenVerse (exp 2) leads to a 4% relative improvement in WER compared to the baseline ASR model (exp 1). For models trained on a single task (exp 3a-c), ASR results remain similar except for SCD. On the *CallHome* dataset (Tab. 5), the multitask model with all tokens yields a 7.7% relative improvement. Overall, the results on both datasets indicate that the all-tasks TokenVerse improves ASR performance.

Named-Entity Recognition As expected, compared to evaluating BERT-NER on reference text, a significant degradation is observed when evaluated on hypothesis (Tab. 3) due to ASR errors (Ghannay et al., 2018). In exact-match, on both the *DefinedAI* (Tab. 3) and *CallHome* (Tab. 5) test sets, the all-tasks TokenVerse outperforms the baseline BERT-NER models trained on their respective datasets and evaluated on hypothesis in F1 score. This is not the case for soft-match evaluation on the *DefinedAI* test set, where the F1 score is similar. This degradation is mostly attributed to the incorrect prediction of [/NE] tag by the baseline, resulting in only a partial match of the named entity words leading to increase in false positives. The absolute F1 score is low on the *CallHome* dataset due to higher ASR errors on named entities, attributed to their low repetition in the training text (see Tab. 1).

Speaker Change Detection On the *DefinedAI* (Tab. 4), including all tasks in TokenVerse outperforms the baseline PyAnnote model in time-based evaluations. Interestingly, models trained for single-task SCD perform better than the all-tasks model in terms of F1, but show similar results for Coverage-Purity based F1. Upon closer scrutiny, we found that including [ENDP] delays the prediction for [SCD] tokens, causing the hypothesis timestamps of these tokens to fall outside the tolerance window (250ms). Increasing the tolerance window further improves the F1 for both models, with a much higher rate of increase for the all-tasks model. This observation is reinforced in the text-

[6] https://github.com/k2-fsa/icefall/tree/master/egs/librispeech/ASR/zipformer

Table 4: [SCD] and [ENDP] time-based evaluation. FA: false alarm; MS: missed speech; DER: detection error rate. †F1-score computed from the Coverage-Purity. ‡single-task model per task, i.e., SCD and ENDP.

Exp	Model	SCD		EndPointing			
		F1	CP-F1†	F1	FA	MS	DER
b-1/2)	PyAnnote	69.6	92.2	73.5	**1.1**	8.5	9.6
2)	all-tasks	79.7	**97.7**	**85.7**	4.7	**1.4**	6.1
3-a/c)	single‡	**87.5**	97.6	84.1	1.9	2.0	**3.9**

based F1 score, where the all-tasks model achieves an F1 score of 90.3% compared to 88.5% from the single-[SCD] model. On the *CallHome* (Tab. 5), the all-tasks model outperforms the PyAnnote baseline. These evaluations suggest that excluding [SCD] from TokenVerse is preferable for precise speaker change timestamps, while including all tasks improves speaker-attributed text segmentation.

Endpointing In text-based evaluation on the *DefinedAI* (Tab. 3) and *CallHome* (Tab. 5) test sets, the all-tasks TokenVerse outperforms the BERT-ENDP models trained on respective datasets. Additionally, on the *DefinedAI* dataset, we evaluate the BERT-ENDP model on both reference and hypothesis to understand the effect of ASR errors on [ENDP] token prediction. Interestingly, we do not observe a significant degradation when evaluating on the hypothesis compared to the reference. This suggests that errors introduced by ASR may not drastically affect the semantic meaning of the sentences. In time-based evaluation on the *DefinedAI* test set (Tab 4), the all-tasks model outperforms the baseline PyAnnote segmentation model. However, single-task ENDP is better than including all tasks in DER due to lower false alarms.

TokenVerse Ablation Results In ASR, we observed degradation for all ablation experiments (see §2.3), with the largest relative degradation of 2.4% in WER when [ENDP] was removed. Transfer learning on any of the 3 tasks do not degrade ASR performance further. The text-based evaluations of other tasks on *DefinedAI* are reported in Figure 2; absolute change is calculated from the all-tasks model. Removing a task adversely affects other tasks. Specifically, for SCD and endpointing, [NE] removal has the least impact on performance. Learning it afterward either improves or maintain their performance, indicating a stronger correlation between these tasks than with NER; supported by the degradation in [SCD] performance when [ENDP] is removed. Task transfer on [ENDP] degrades the performance further, possibly due to

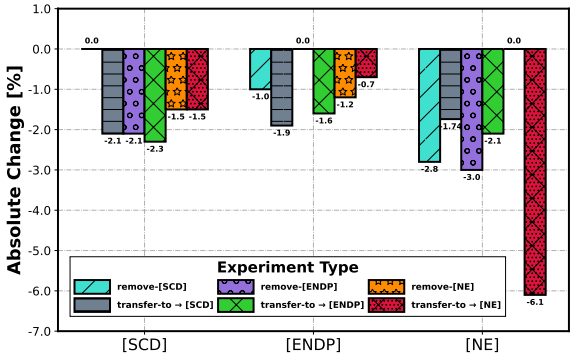

Figure 2: Absolute changes in text-based evaluation w.r.t all-tasks TokenVerse in @F1. We either remove a task, e.g., remove-[NE], or transfer to the removed task, e.g., transfer-to →[NE]. Note that all-tasks TokenVerse performs better in all scenarios.

Table 5: F1-score and WER for CallHome Eval set on different tasks with TokenVerse. †time-based F1 score. ‡baselines are computed with PyAnnote for SCD or with fine-tuned BERT on ENDP and NER (exact-match).

Exp	ASR WER (↓)	SCD† F1 (↑)	ENDP F1 (↑)	NER F1 (↑)
baselines‡	24.6	91.7	55.9	27.4
all-tasks	**22.7**	**92.5**	**73.3**	**30.6**

confusion during prediction caused by the insertion of the token before [SCD] during training. Transfer to NER shows relatively large degradation compared to other tasks, likely because the model must predict both [NE] and [/NE] accurately. This suggests that tasks encoded with multiple tokens may not transfer as effectively as those encoded with a single token.

Overall, all-tasks TokenVerse outperforms specialized models for each task and single-task models suggesting that additional tasks improve each other. See sample outputs in appendix B.

6 Conclusions

In this paper, we show the effectiveness of a token-based multitask model on speech and NLP using XLSR-Transducer as our ASR model, termed TokenVerse. Alongside ASR, speaker change detection, endpointing and named entity recognition are considered. Results on 2 datasets show that our approach improves ASR performance while outperforming strong task-specific baselines. Ablation experiments suggest that multitask training across different domains can enhance performance on all tasks. Our approach offers flexibility for extension to numerous tasks across various domains.

Limitations

One major limitation of our work is the restricted size of the datasets used in our experiments. The scope of our research involves performing multiple tasks on conversational audios, making it challenging to find an open-source dataset that provides annotations for all the considered tasks. Another limitation is that we do not consider multiple entity types, instead assuming a single entity type, which limits the usability of our proposed model in scenarios where entity type predictions are required.

Acknowledgements

This work was supported by the Idiap & Uniphore collaboration project. Part of the work was also support supported by EU Horizon 2020 project ELOQUENCE[7] (grant number 101070558).

References

Hervé Bredin. 2023. pyannote. audio 2.1 speaker diarization pipeline: principle, benchmark, and recipe. In *24th INTERSPEECH Conference*, pages 1983–1987. ISCA.

Hervé Bredin, Claude Barras, et al. 2017. Speaker change detection in broadcast tv using bidirectional long short-term memory networks. In *Interspeech 2017*. ISCA.

Bredin, Hervé. 2017. pyannote.metrics: A toolkit for reproducible evaluation, diagnostic, and error analysis of speaker diarization systems. In *Proc. Interspeech 2017*, pages 3587–3591. ISCA.

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

Shuo-Yiin Chang, Rohit Prabhavalkar, Yanzhang He, Tara N Sainath, and Gabor Simko. 2019. Joint endpointing and decoding with end-to-end models. In *ICASSP*, pages 5626–5630. IEEE.

Yi-Chen Chen, Shu-wen Yang, Cheng-Kuang Lee, Simon See, and Hung-yi Lee. 2021. Speech representation learning through self-supervised pretraining and multi-task finetuning. *arXiv preprint arXiv:2110.09930*.

Ido Cohn, Itay Laish, Genady Beryozkin, Gang Li, Izhak Shafran, Idan Szpektor, Tzvika Hartman, Avinatan Hassidim, and Yossi Matias. 2019. Audio de-identification: A new entity recognition task. *arXiv preprint arXiv:1903.07037*.

[7] https://eloquenceai.eu/

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

Samuele Cornell, Jee-weon Jung, Shinji Watanabe, and Stefano Squartini. 2023. One model to rule them all? towards end-to-end joint speaker diarization and speech recognition. *arXiv preprint arXiv:2310.01688*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Sahar Ghannay, Antoine Caubriere, Yannick Esteve, Antoine Laurent, and Emmanuel Morin. 2018. End-to-end named entity extraction from speech. *arXiv preprint arXiv:1805.12045*.

Mohammadreza Ghodsi, Xiaofeng Liu, James Apfel, Rodrigo Cabrera, and Eugene Weinstein. 2020. Rnn-transducer with stateless prediction network. In *ICASSP*, pages 7049–7053. IEEE.

Alex Graves. 2012. Sequence transduction with recurrent neural networks. *arXiv preprint arXiv:1211.3711*.

Naoyuki Kanda, Jian Wu, Yu Wu, Xiong Xiao, Zhong Meng, Xiaofei Wang, Yashesh Gaur, Zhuo Chen, Jinyu Li, and Takuya Yoshioka. 2022. Streaming multi-talker asr with token-level serialized output training. *arXiv preprint arXiv:2202.00842*.

Fangjun Kuang, Liyong Guo, Wei Kang, Long Lin, Mingshuang Luo, Zengwei Yao, and Daniel Povey. 2022. Pruned rnn-t for fast, memory-efficient asr training. *arXiv preprint arXiv:2206.13236*.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*.

Shashi Kumar, Srikanth Madikeri, Nigmatulina Iuliia, Esaú VILLATORO-TELLO, Petr Motlicek, Karthik Pandia D S, S. Pavankumar Dubagunta, and Aravind Ganapathiraju. 2024. Multitask speech recognition and speaker change detection for unknown number of speakers. In *Proceedings of the 49th IEEE International Conference on Acoustics, Speech, & Signal Processing (ICASSP) 2024*.

Jing Li, Aixin Sun, Jianglei Han, and Chenliang Li. 2020. A survey on deep learning for named entity recognition. *IEEE transactions on knowledge and data engineering*, 34(1):50–70.

Jonathan Mamou, David Carmel, and Ron Hoory. 2006. Spoken document retrieval from call-center conversations. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 51–58.

Ivan Medennikov, Maxim Korenevsky, Tatiana Prisyach, Yuri Khokhlov, Mariya Korenevskaya, Ivan Sorokin, Tatiana Timofeeva, Anton Mitrofanov, Andrei Andrusenko, Ivan Podluzhny, et al. 2020. Target-speaker voice activity detection: a novel approach for multi-speaker diarization in a dinner party scenario. *arXiv preprint arXiv:2005.07272*.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*.

Tae Jin Park, Naoyuki Kanda, Dimitrios Dimitriadis, Kyu J Han, Shinji Watanabe, and Shrikanth Narayanan. 2022. A review of speaker diarization: Recent advances with deep learning. *Computer Speech & Language*, 72:101317.

Alexis Plaquet and Hervé Bredin. 2023. Powerset multi-class cross entropy loss for neural speaker diarization. In *INTERSPEECH 2023*, pages 3222–3226.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *International Conference on Machine Learning*, pages 28492–28518. PMLR.

Antoine Raux and Maxine Eskenazi. 2008. Optimizing endpointing thresholds using dialogue features in a spoken dialogue system. In *Proceedings of the 9th SIGdial Workshop on Discourse and Dialogue*, pages 1–10.

Morteza Saberi, Omar Khadeer Hussain, and Elizabeth Chang. 2017. Past, present and future of contact centers: a literature review. *Business Process Management Journal*, 23(3):574–597.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725.

Laurent El Shafey, Hagen Soltau, and Izhak Shafran. 2019. Joint speech recognition and speaker diarization via sequence transduction. *arXiv preprint arXiv:1907.05337*.

Shu wen Yang, Po-Han Chi, et al. 2021. SUPERB: Speech Processing Universal PERformance Benchmark. In *Proc. Interspeech 2021*, pages 1194–1198.

Jian Wu, Naoyuki Kanda, Takuya Yoshioka, Rui Zhao, Zhuo Chen, and Jinyu Li. 2023. t-sot fnt: Streaming multi-talker asr with text-only domain adaptation capability. *arXiv preprint arXiv:2309.08131*.

Yihan Wu, Soumi Maiti, Yifan Peng, Wangyou Zhang, Chenda Li, Yuyue Wang, Xihua Wang, Shinji Watanabe, and Ruihua Song. 2024. Speechcomposer: Unifying multiple speech tasks with prompt composition. *arXiv preprint arXiv:2401.18045*.

Wei Xia, Han Lu, Quan Wang, Anshuman Tripathi, Yiling Huang, Ignacio Lopez Moreno, and Hasim Sak. 2022. Turn-to-diarize: Online speaker diarization constrained by transformer transducer speaker turn detection. In *ICASSP*, pages 8077–8081. IEEE.

Yi Xu, Hai Zhao, and Zhuosheng Zhang. 2021. Topic-aware multi-turn dialogue modeling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14176–14184.

Hemant Yadav, Sreyan Ghosh, Yi Yu, and Rajiv Ratn Shah. 2020. End-to-end named entity recognition from english speech. *arXiv preprint arXiv:2005.11184*.

Yicheng Zou, Lujun Zhao, Yangyang Kang, Jun Lin, Minlong Peng, Zhuoren Jiang, Changlong Sun, Qi Zhang, Xuanjing Huang, and Xiaozhong Liu. 2021. Topic-oriented spoken dialogue summarization for customer service with saliency-aware topic modeling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14665–14673.

Juan Pablo Zuluaga-Gomez, Zhaocheng Huang, Xing Niu, Rohit Paturi, Sundararajan Srinivasan, Prashant Mathur, Brian Thompson, and Marcello Federico. 2023. End-to-end single-channel speaker-turn aware conversational speech translation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7255–7274.

A Metrics & Evaluation Protocol

Named-Entity Recognition *Exact-Match:* Let $P = \{P_1, P_2, \ldots, P_n\}$ be the set of predicted entities, and $A = \{A_1, A_2, \ldots, A_n\}$ be the set of actual entities, where each P_i and A_i is accompanied by its corresponding `[NE]-[/NE]` tokens (See Fig.1). Thus, an entity P_i is considered correctly identified if and only if: $\forall i \in \{1, 2, \ldots, n\}, P_i = A_i$, including the tokens. Unmatched pairs of tokens in reference are considered false negative. Similarly, unmatched open or close tokens in hypothesis are considered false positive. *Soft-Match:* in this case we only count for the paired sets of `[NE]-[/NE]` tokens without considering if the predicted entity value P_i was correctly transcribed. After obtaining each pair and unmatched tokens, we evaluate NER with F1-score.

Speaker Change Detection In text-based evaluation, we align the reference and hypothesis using edit-distance. For each occurrence of the `[SCD]` token in the reference, matching with the same token in the hypothesis counts as True Positive; else, False Negative. Unmatched tokens in the hypothesis are considered False Positive. F1 score is calculated by standard definitions. In time-based evaluation, we obtain the timestamps where `[SCD]` tokens are predicted in the hypothesis. We calculate F1 score (Kumar et al., 2024), using a collar of 250ms during timestamp matching, following common practice in speaker diarization literature (Park et al., 2022). Additionally, segment coverage, purity (Bredin et al., 2017), and their F1 score are also reported. We use `pyannote.metrics` (Bredin, Hervé, 2017) to compute all time-based metrics.

B Sample output from TokenVerse

Reference: hello thank you for calling *geico insurance* my name is *alexa* how may i help you today

ASR only model: hello thank you for calling *geico insurance* my name is *allesa* how may i help you today

TokenVerse model: hello thank you for calling `[NE]` *geico insurance* `[/NE]` my name is `[NE]` ***alexa*** `[/NE]` how may i help you today